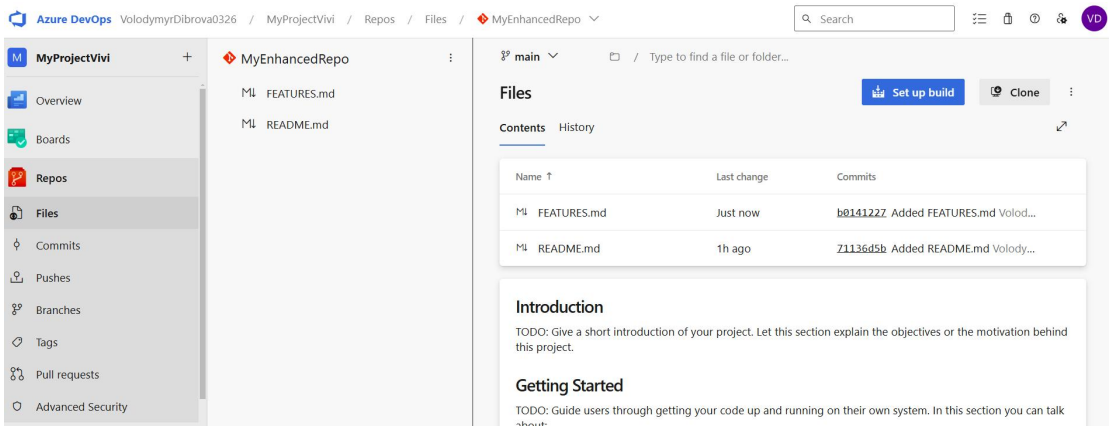
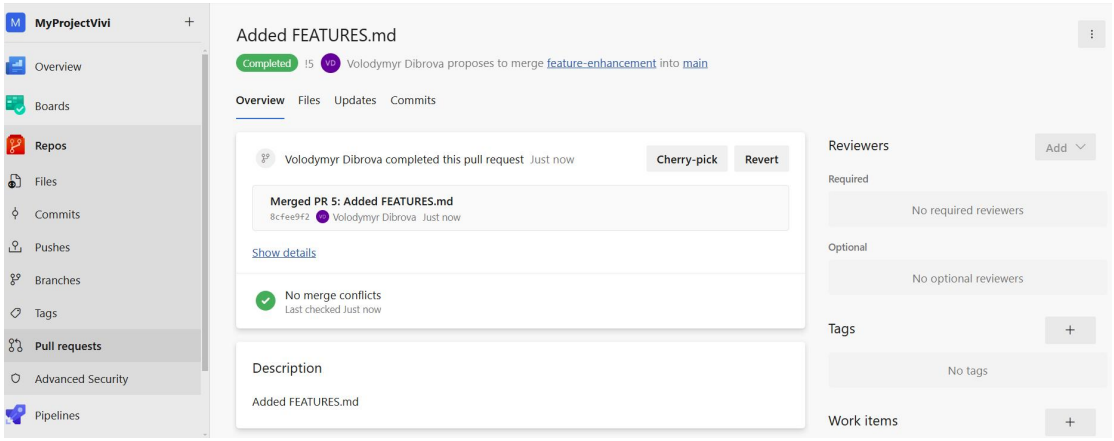
My url repo in DevAzure : https://dev.azure.com/VolodymyrDibrova0326/MyProjectVivi/_git/fullstack-azure-app

Practical Task 1: Create and Initialize a New Repository with a README
Practical Task 2: Implement Branching and Merging



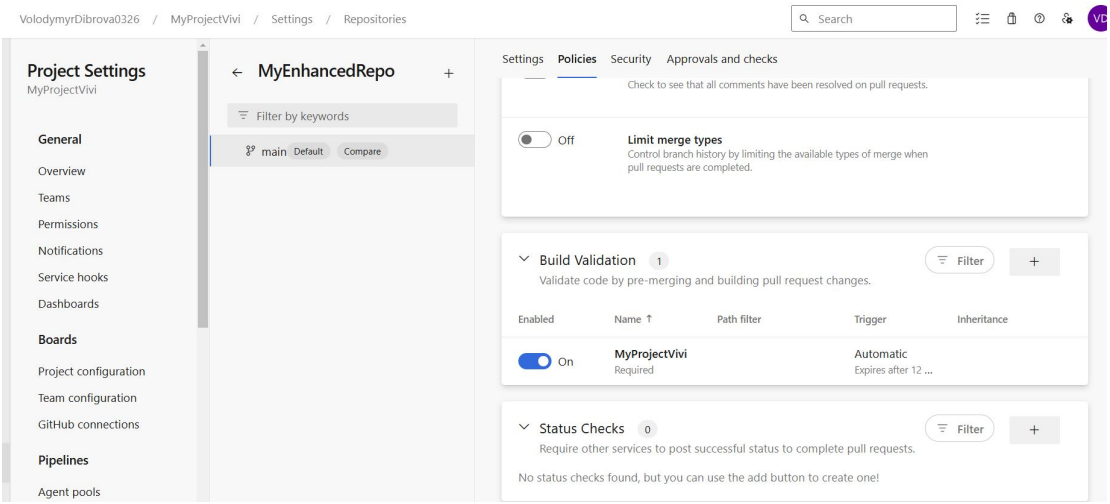


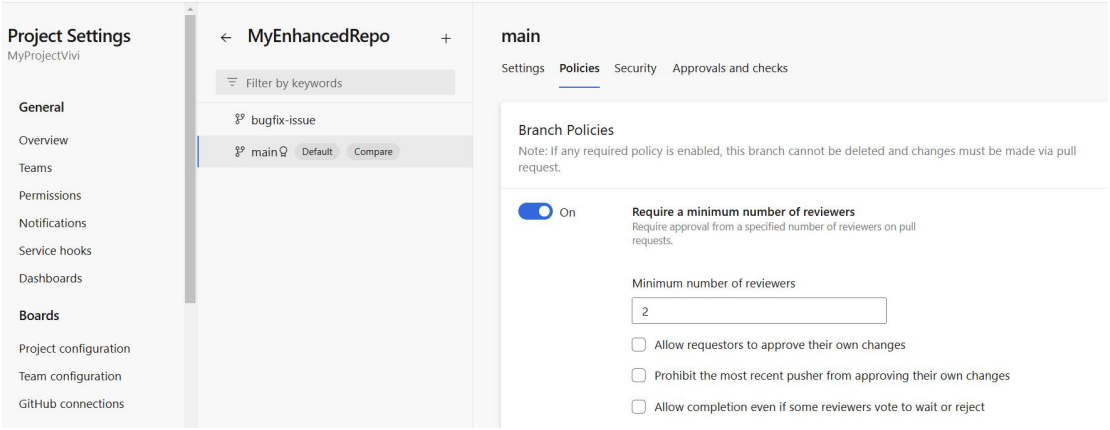
Practical Task 3: Set Up Branch Policies and Code Review
Requirements:

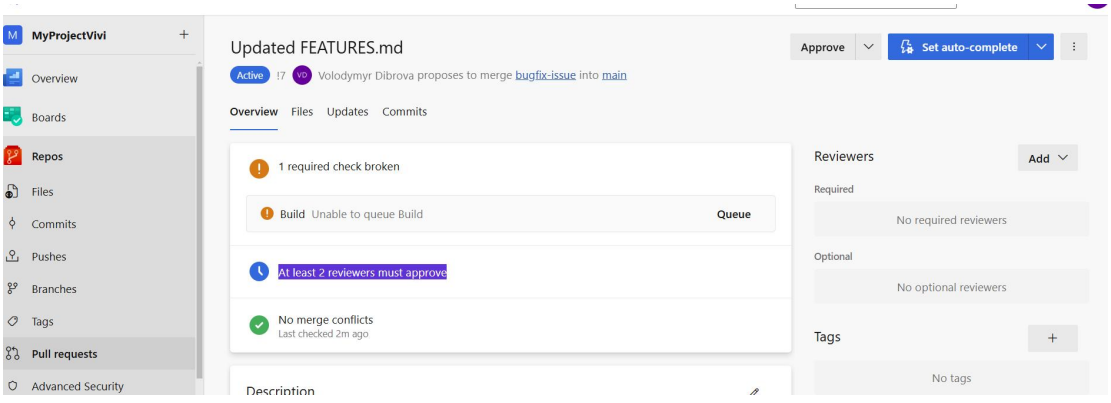
- Configure branch policies for the **main** branch in Azure Repos to require at least one reviewer for pull requests and to enforce a minimum number of successful builds (you can simulate this by using a mock build status if needed).
- Create a new branch named **bugfix-issue** from the **main** branch and make changes to the **FEATURES.md** file to address a hypothetical bug (e.g., update a feature description).
- Commit the changes and push the **bugfix-issue** branch to the remote repository.
- Create a pull request from **bugfix-issue** to **main**, ensuring that the pull request meets the branch policies set earlier.
- Invite at least one team member or colleague to review the pull request and provide feedback.
- Address any comments or requested changes from the reviewer and update the pull request accordingly.

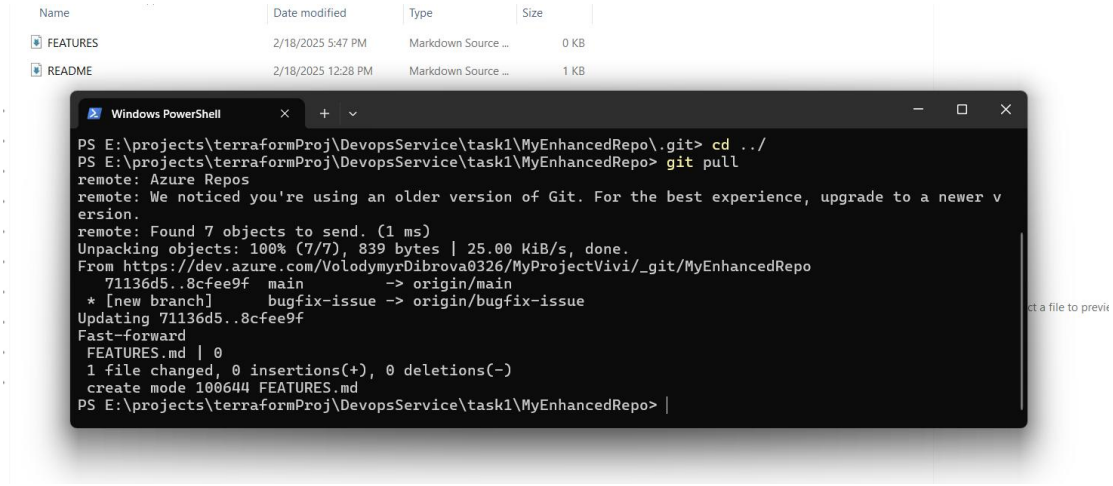
- Merge the pull request into the **main** branch once it has been approved.

1.Configured and setted a 2 number of reviewers



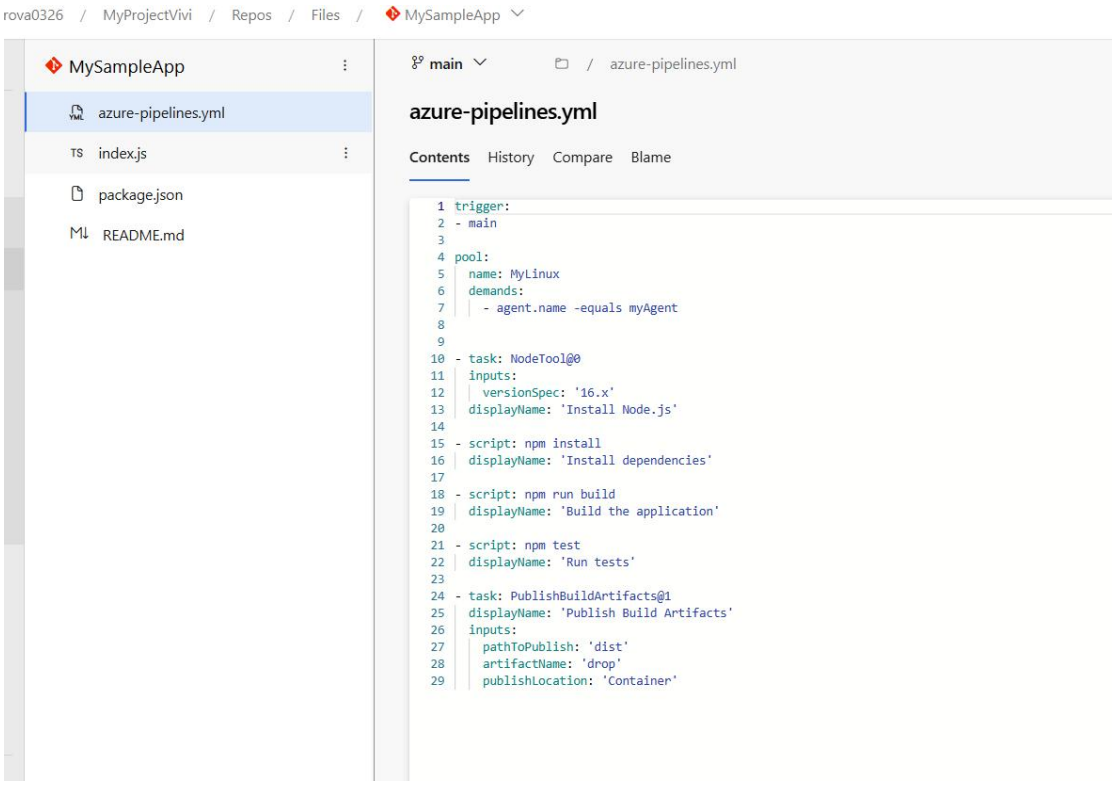


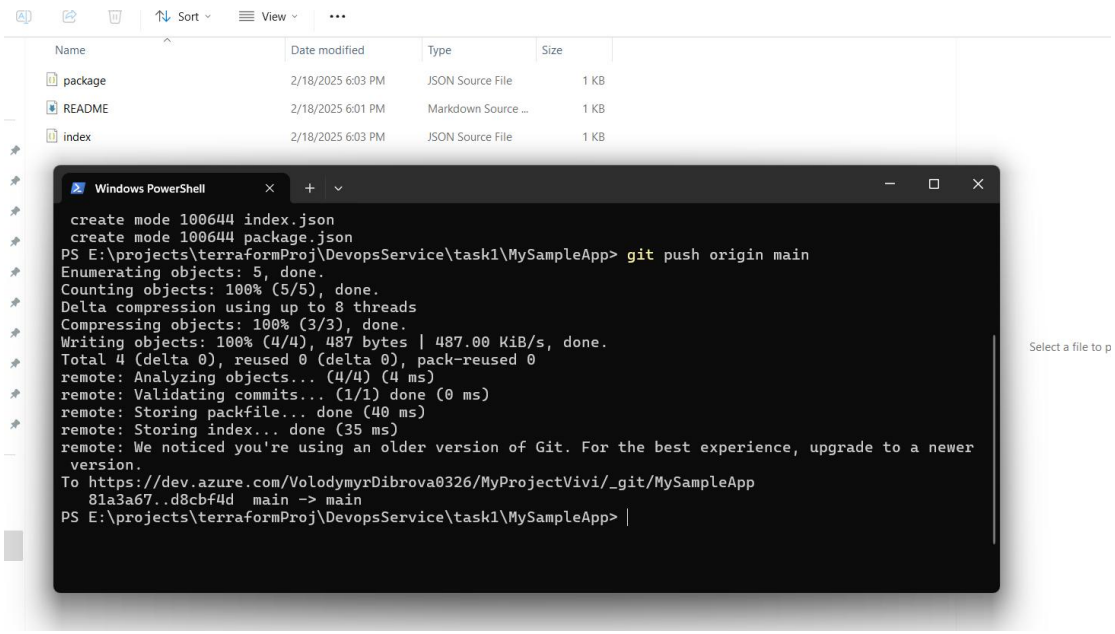


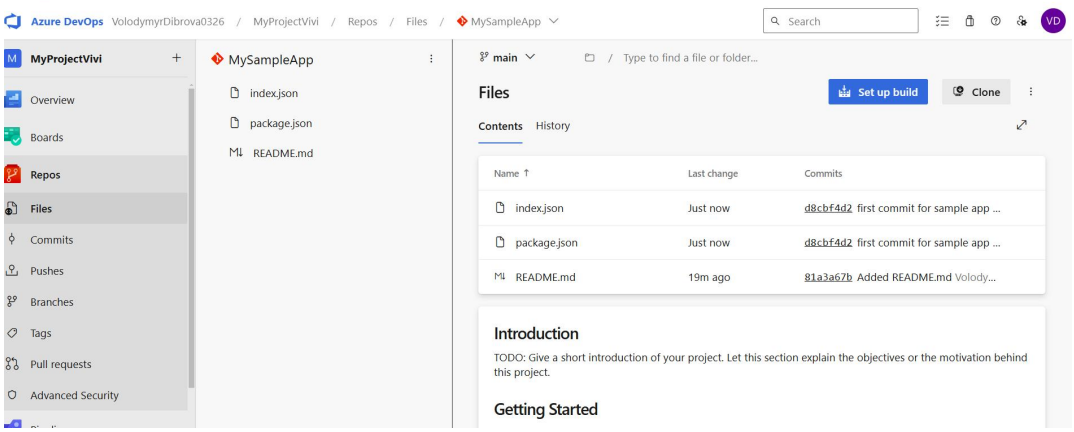


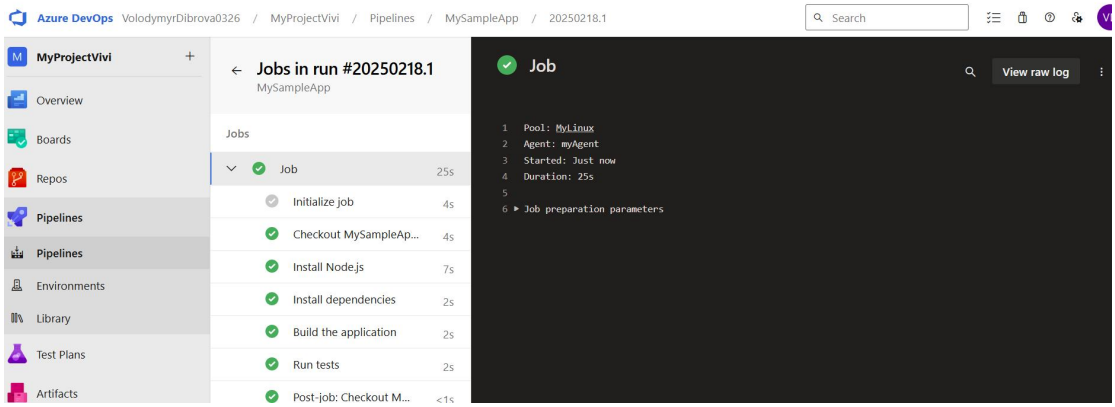
Practical Task 4: Create a Simple Build Pipeline in Azure DevOps
Requirements:
1. Log in to Azure DevOps:
a. Access your Azure DevOps organization and navigate to your project.
2. Create a Git Repository:
a. In Azure Repos, create a new Git repository named MySampleApp.
b. Clone the repository to your local machine and add a simple application (e.g., a
basic .NET Core or Node.js application).
3. Create a Build Pipeline:
a. Navigate to Azure Pipelines and create a new pipeline.
b. Select the Azure Repos Git repository as the source.c. Use the YAML pipeline editor to define the pipeline, including the following steps:
dotnet restore (for .NET) or npm install (for Node.js) to restore dependencies.
dotnet build (for .NET) or npm run build (for Node.js) to build the application.
dotnet test (for .NET) or a testing command for Node.js to run unit tests.
4. Run the Pipeline:
a. Save and run the pipeline.
b. Verify that the build pipeline completes successfully and check the build logs for

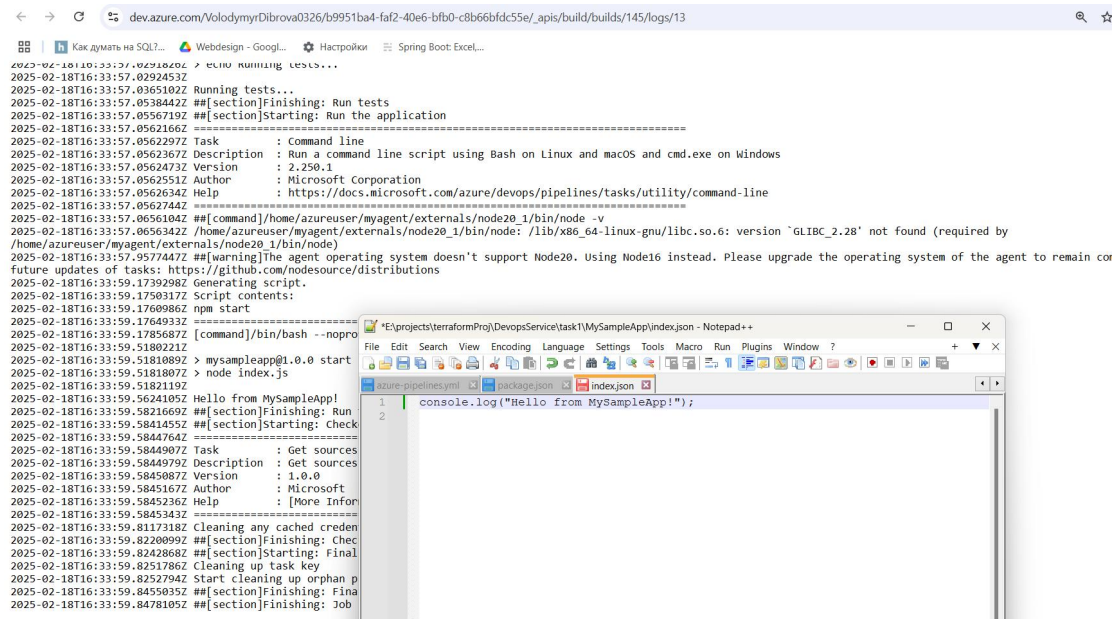
any
error


rova0326 / MyProjectVivi / Repos / Files / MySampleApp
MySampleApp
azure-pipelines.yml
index.js
package.json
README.md
main / azure-pipelines.yml
azure-pipelines.yml
Contents History Compare Blame
1 trigger:
2 - main
3
4 pool:
5 name: MyLinux
6 demands:
7 - agent.name -equals myAgent
8
9
10 - task: NodeTool@0
11 inputs:
12 versionSpec: '16.x'
13 displayName: 'Install Node.js'
14
15 - script: npm install
16 displayName: 'Install dependencies'
17
18 - script: npm run build
19 displayName: 'Build the application'
20
21 - script: npm test
22 displayName: 'Run tests'
23
24 - task: PublishBuildArtifacts@1
25 displayName: 'Publish Build Artifacts'
26 inputs:
27 pathToPublish: 'dist'
28 artifactName: 'drop'
29 publishLocation: 'Container'

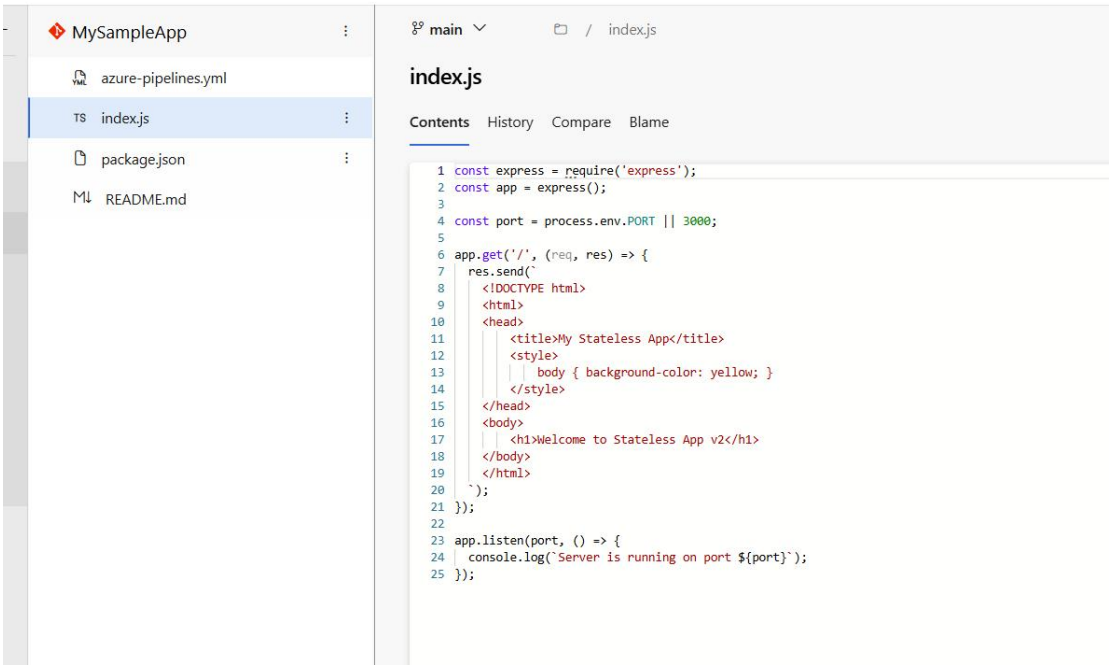

Sort
View
Name
Date modified
Type
Size
package
2/18/2025 6:03 PM
JSON Source File
1 KB
README
2/18/2025 6:01 PM
Markdown Source ...
1 KB
index
2/18/2025 6:03 PM
JSON Source File
1 KB
Windows PowerShell
create mode 100644 index.json
create mode 100644 package.json
PS E:\projects\terraformProj\DevopsService\task1\MySampleApp> git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 487 bytes | 487.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Analyzing objects... (4/4) (4 ms)
remote: Validating commits... (1/1) done (0 ms)
remote: Storing packfile... done (40 ms)
remote: Storing index... done (35 ms)
remote: We noticed you're using an older version of Git. For the best experience, upgrade to a newer version.
To https://dev.azure.com/VolodymyrDibrova0326/MyProjectVivi/_git/MySampleApp
81a3a67..d8cbf4d main -> main
PS E:\projects\terraformProj\DevopsService\task1\MySampleApp>
Select a file to p

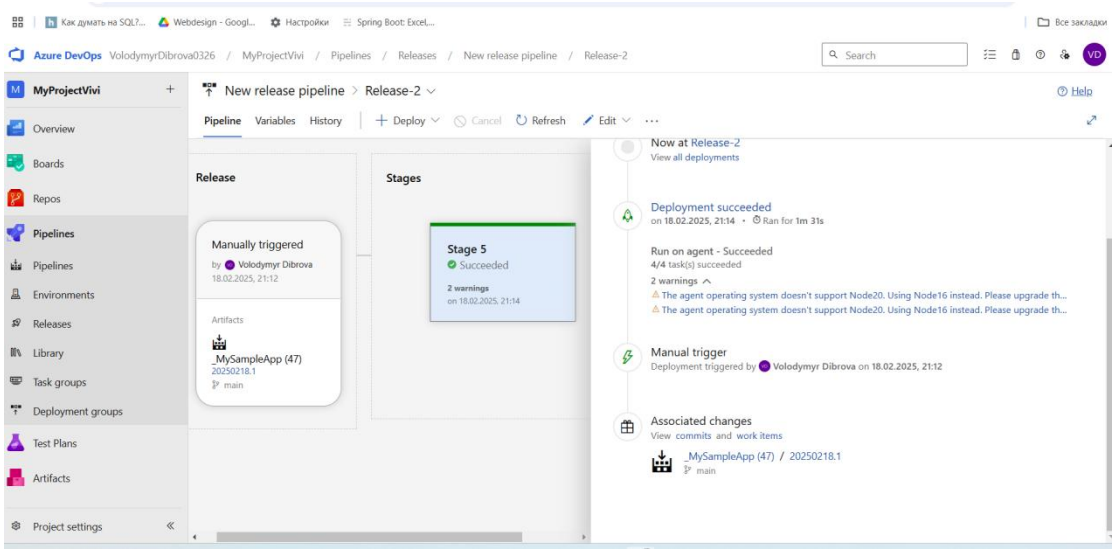

Azure DevOps VolodymyrDibrova0326 / MyProjectVivi / Repos / Files / MySampleApp
Search
MyProjectVivi
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags
Pull requests
Advanced Security
MySampleApp
index.json
package.json
README.md
main
Type to find a file or folder...
Files
Set up build
Clone
Contents
History
Name
Last change
Commits
index.json
Just now
d8cbf4d2 first commit for sample app ...
package.json
Just now
d8cbf4d2 first commit for sample app ...
README.md
19m ago
81a3a67b Added README.md Volody...
Introduction
TODO: Give a short introduction of your project. Let this section explain the objectives or the motivation behind this project.
Getting Started

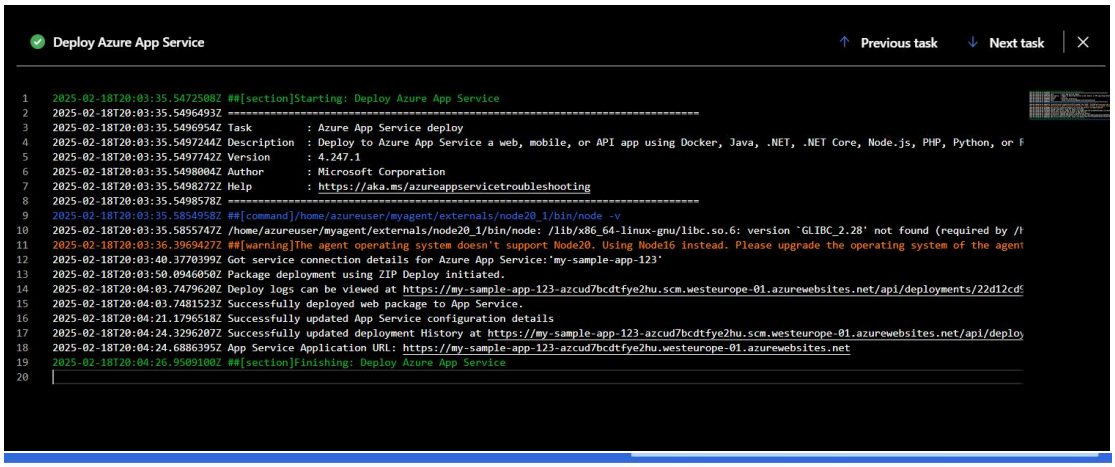

Azure DevOps VolodymyrDibrova0326 / MyProjectVivi / Pipelines / MySampleApp / 20250218.1
Search
MyProjectVivi
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Library
Test Plans
Artifacts
Jobs in run #20250218.1
MySampleApp
Jobs
Job 25s
Initialize job 4s
Checkout MySampleAp... 4s
Install Node.js 7s
Install dependencies 2s
Build the application 2s
Run tests 2s
Post-job: Checkout M... <1s
Job
View raw log
1 Pool: MyLinux
2 Agent: myAgent
3 Started: Just now
4 Duration: 25s
5
6 ▶ Job preparation parameters

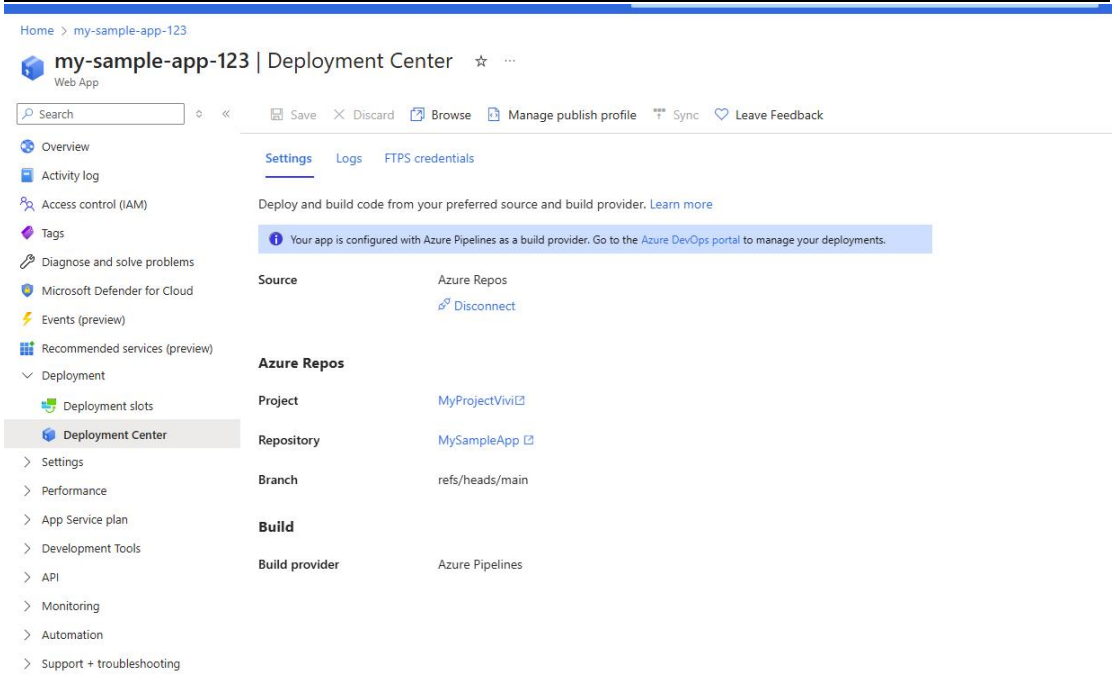

dev.azure.com/VolodymyrDibrova0326/b9951ba4-faf2-40e6-bfb0-c8b66bfdc55e/_apis/build/builds/145/logs/13
2025-02-18T16:33:57.0292453Z
2025-02-18T16:33:57.0365102Z Running tests...
2025-02-18T16:33:57.0538442Z ##[section]Finishing: Run tests
2025-02-18T16:33:57.0556719Z ##[section]Starting: Run the application
2025-02-18T16:33:57.0562297Z Task : Command line
2025-02-18T16:33:57.0562367Z Description : Run a command line script using Bash on Linux and macOS and cmd.exe on Windows
2025-02-18T16:33:57.0562473Z Version : 2.250.1
2025-02-18T16:33:57.0562551Z Author : Microsoft Corporation
2025-02-18T16:33:57.0562634Z Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/utility/command-line
2025-02-18T16:33:57.0656104Z ##[command]/home/azureuser/myagent/externals/node20_1/bin/node -v
2025-02-18T16:33:57.0656342Z /home/azureuser/myagent/externals/node20_1/bin/node: /lib/x86_64-linux-gnu/libc.so.6: version `GLIBC_2.28' not found (required by /home/azureuser/myagent/externals/node20_1/bin/node)
2025-02-18T16:33:57.9577447Z ##[warning]The agent operating system doesn't support Node20. Using Node16 instead. Please upgrade the operating system of the agent to remain cor
future updates of tasks: https://github.com/nodesource/distributions
2025-02-18T16:33:59.1739298Z Generating script.
2025-02-18T16:33:59.1750317Z Script contents:
2025-02-18T16:33:59.1760986Z npm start
2025-02-18T16:33:59.1785687Z [command]/bin/bash --nopro
2025-02-18T16:33:59.5180289Z > mysampleapp@1.0.0 start
2025-02-18T16:33:59.5181807Z > node index.js
2025-02-18T16:33:59.5624105Z Hello from MySampleApp!
2025-02-18T16:33:59.5821669Z ##[section]Finishing: Run
2025-02-18T16:33:59.5841455Z ##[section]Starting: Check
2025-02-18T16:33:59.5844907Z Task : Get sources
2025-02-18T16:33:59.5844979Z Description : Get sources
2025-02-18T16:33:59.5845087Z Version : 1.0.0
2025-02-18T16:33:59.5845167Z Author : Microsoft
2025-02-18T16:33:59.5845236Z Help : [More Infor
2025-02-18T16:33:59.8117318Z Cleaning any cached creden
2025-02-18T16:33:59.8220099Z ##[section]Finishing: Chec
2025-02-18T16:33:59.8242868Z ##[section]Starting: Final
2025-02-18T16:33:59.8251786Z Cleaning up task key
2025-02-18T16:33:59.8252794Z Start cleaning up orphan p
2025-02-18T16:33:59.8455035Z ##[section]Finishing: Fina
2025-02-18T16:33:59.8478105Z ##[section]Finishing: Job
*E:\projects\terraformProj\DevopsService\task1\MySampleApp\index.json - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
azure-pipelines.yml
package.json
index.json
1 console.log("Hello from MySampleApp!");
2

Practical Task 5: Set Up Continuous Deployment (CD) to Azure Web App
Requirements:
1. Extend the Build Pipeline:
a. Open the existing build pipeline created in Task 4.
b. Add a new stage for deployment after the build stage.
2. Create an Azure Web App:
a. In the Azure portal, manually create an Azure Web App or use an ARM template to
create it.
3. Configure Release Pipeline:
a. Create a release pipeline in Azure DevOps that triggers automatically after a
successful build.
b. Add an Azure Web App deployment task to the release pipeline, specifying the Azure
subscription and the Web App name.
4. Deploy the Application:
a. Save and run the release pipeline.
b. Verify that the application is successfully deployed and accessible via the web



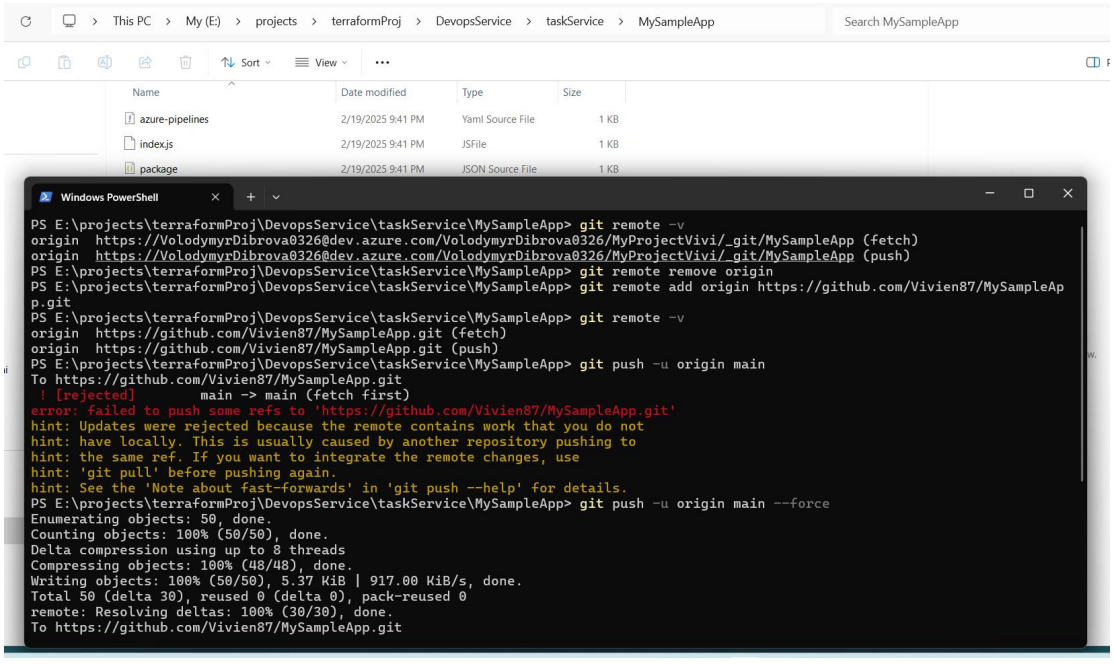

Azure DevOps VolodymyrDibrova0326 / MyProjectVivi / Pipelines / Releases / New release pipeline / Release-2
MyProjectVivi
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
Library
Task groups
Deployment groups
Test Plans
Artifacts
Project settings
New release pipeline > Release-2
Pipeline Variables History
Deploy Cancel Refresh Edit
Release
Manually triggered
by Volodymyr Dibrova
18.02.2025, 21:12
Artifacts
_MySampleApp (47)
20250218.1
main
Stages
Stage 5
Succeeded
2 warnings
on 18.02.2025, 21:14
Now at Release-2
View all deployments
Deployment succeeded
on 18.02.2025, 21:14 · Ran for 1m 31s
Run on agent - Succeeded
4/4 task(s) succeeded
2 warnings
The agent operating system doesn't support Node20. Using Node16 instead. Please upgrade th...
The agent operating system doesn't support Node20. Using Node16 instead. Please upgrade th...
Manual trigger
Deployment triggered by Volodymyr Dibrova on 18.02.2025, 21:12
Associated changes
View commits and work items
_MySampleApp (47) / 20250218.1
main

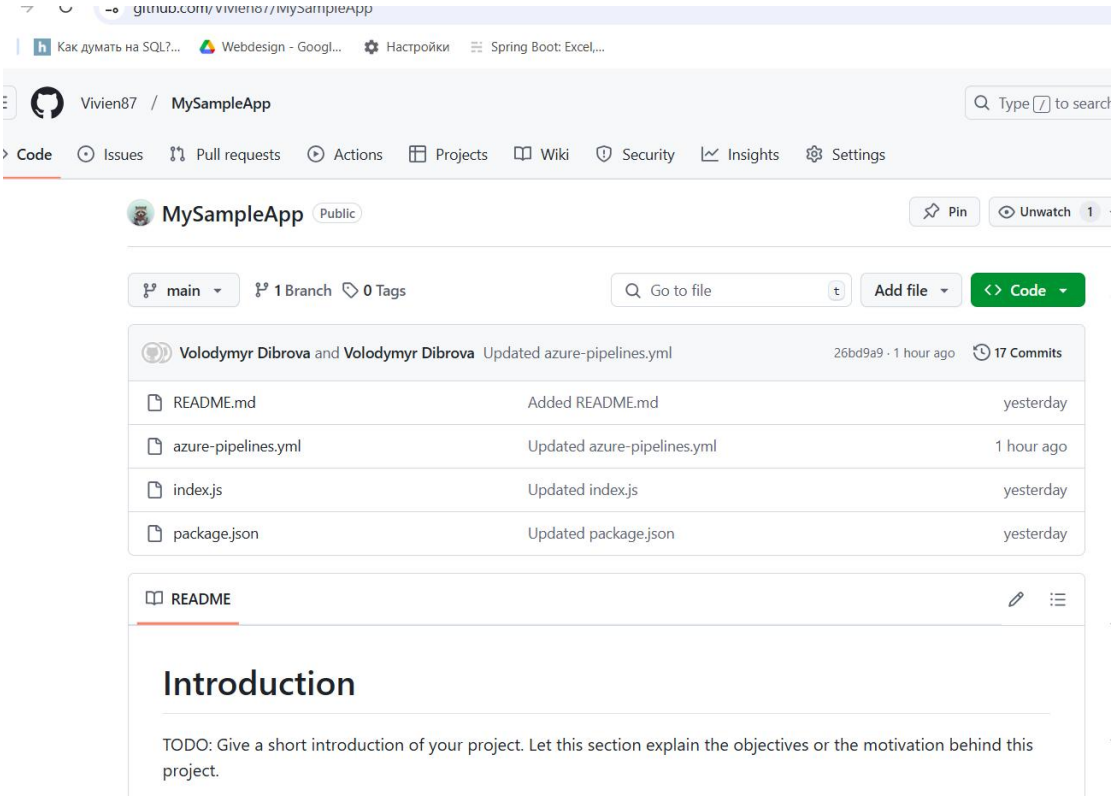

Deploy Azure App Service
Previous task
Next task
2025-02-18T20:03:35.5472508Z ##[section]Starting: Deploy Azure App Service
2025-02-18T20:03:35.5496493Z ==
2025-02-18T20:03:35.5496954Z Task : Azure App Service deploy
2025-02-18T20:03:35.5497244Z Description : Deploy to Azure App Service a web, mobile, or API app using Docker, Java, .NET, .NET Core, Node.js, PHP, Python, or F
2025-02-18T20:03:35.5497742Z Version : 4.247.1
2025-02-18T20:03:35.5498004Z Author : Microsoft Corporation
2025-02-18T20:03:35.5498272Z Help : https://aka.ms/azureappservicetroubleshooting
2025-02-18T20:03:35.5498578Z ==
2025-02-18T20:03:35.5854958Z ##[command]/home/azureuser/myagent/externals/node20_1/bin/node -v
2025-02-18T20:03:35.5855747Z /home/azureuser/myagent/externals/node20_1/bin/node: /lib/x86_64-linux-gnu/libc.so.6: version `GLIBC_2.28' not found (required by /h
2025-02-18T20:03:36.3969427Z ##[warning]The agent operating system doesn't support Node20. Using Node16 instead. Please upgrade the operating system of the agent
2025-02-18T20:03:40.3770399Z Got service connection details for Azure App Service:'my-sample-app-123'
2025-02-18T20:03:50.0946050Z Package deployment using ZIP Deploy initiated.
2025-02-18T20:04:03.7479620Z Deploy logs can be viewed at https://my-sample-app-123-azcud7bcdtfye2hu.scm.westeurope-01.azurewebsites.net/api/deployments/22d12cd5
2025-02-18T20:04:03.7481523Z Successfully deployed web package to App Service.
2025-02-18T20:04:21.1796518Z Successfully updated App Service configuration details
2025-02-18T20:04:24.3296207Z Successfully updated deployment History at https://my-sample-app-123-azcud7bcdtfye2hu.scm.westeurope-01.azurewebsites.net/api/deploy
2025-02-18T20:04:24.6886395Z App Service Application URL: https://my-sample-app-123-azcud7bcdtfye2hu.westeurope-01.azurewebsites.net
2025-02-18T20:04:26.9509100Z ##[section]Finishing: Deploy Azure App Service

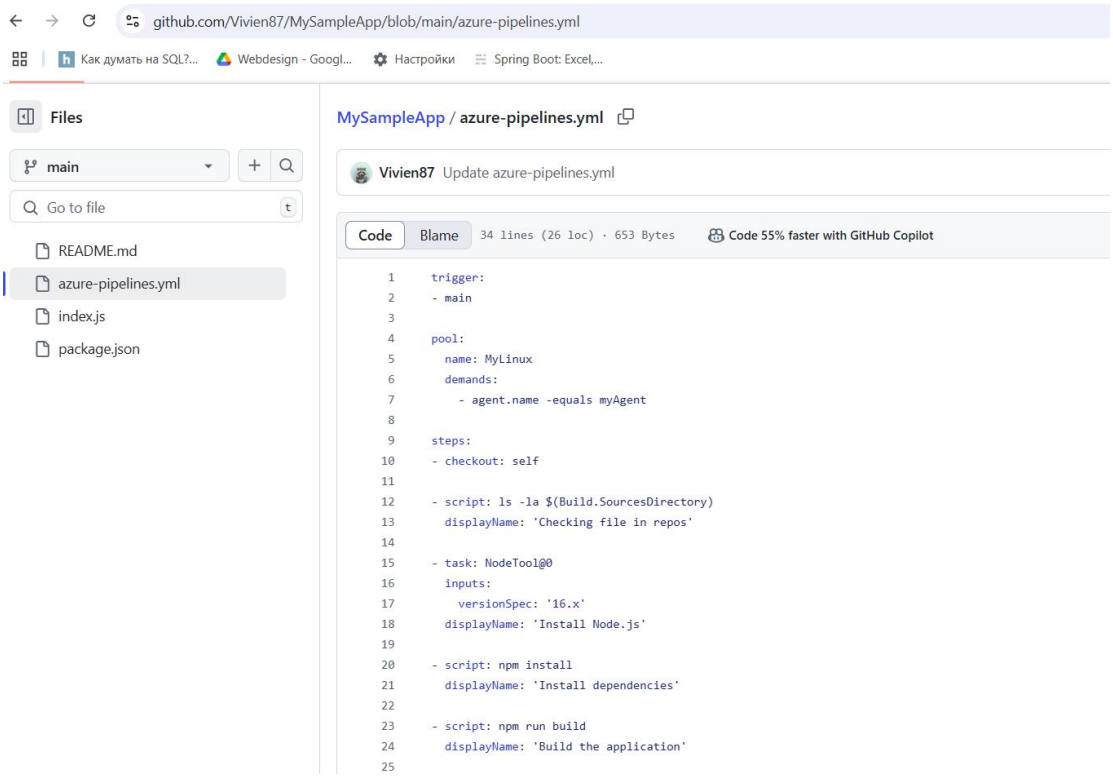

Home > my-sample-app-123
my-sample-app-123 | Deployment Center
Web App
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Microsoft Defender for Cloud
Events (preview)
Recommended services (preview)
Deployment
Deployment slots
Deployment Center
Settings
Performance
App Service plan
Development Tools
API
Monitoring
Automation
Support + troubleshooting
Save Discard Browse Manage publish profile Sync Leave Feedback
Settings Logs FTPS credentials
Deploy and build code from your preferred source and build provider. Learn more
Your app is configured with Azure Pipelines as a build provider. Go to the Azure DevOps portal to manage your deployments.
Source Azure Repos
Disconnect
Azure Repos
Project MyProjectVivi
Repository MySampleApp
Branch refs/heads/main
Build
Build provider Azure Pipelines

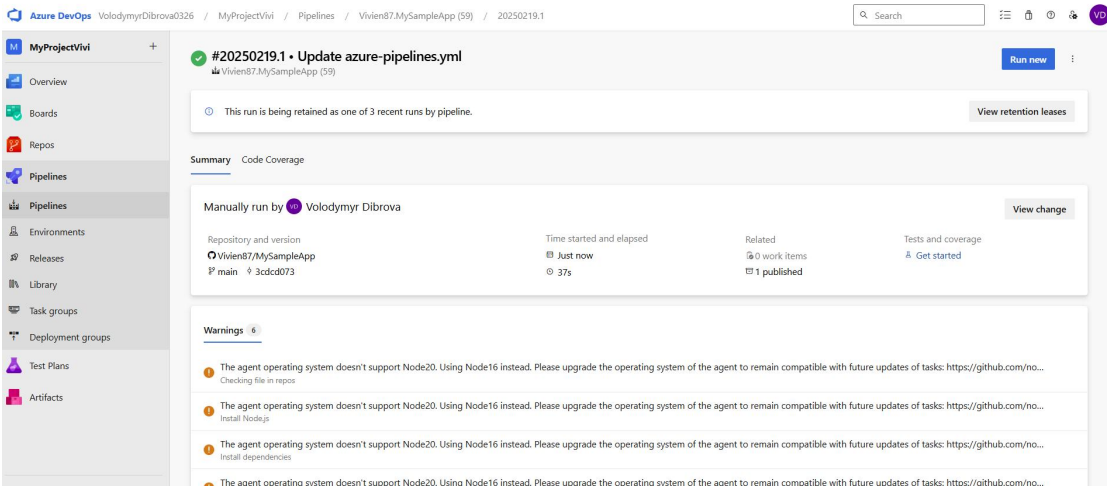
Practical Task 6: Implement CI/CD with GitHub and Azure Pipelines
Requirements:
1. Create a GitHub Repository:
a. Navigate to GitHub and create a new repository named MySampleApp.
2. Push Application Code:
a. Push the application code from Task 4 to the new GitHub repository.
3. Set Up Azure Pipeline:
a. In Azure DevOps, create a new pipeline that uses the GitHub repository as the
source.b. Authenticate with GitHub and select the repository.
4. Define the Pipeline:
a. Use the YAML pipeline editor to define the pipeline with steps to:
i. Build the application.
ii. Run tests.
iii. Deploy to the Azure Web App (similar to Task 5).
5. Enable GitHub Integration:
a. Configure the pipeline to trigger automatically on every push to the main branch.
b. Verify that the pipeline triggers successfully and the application is deployed to Azure.

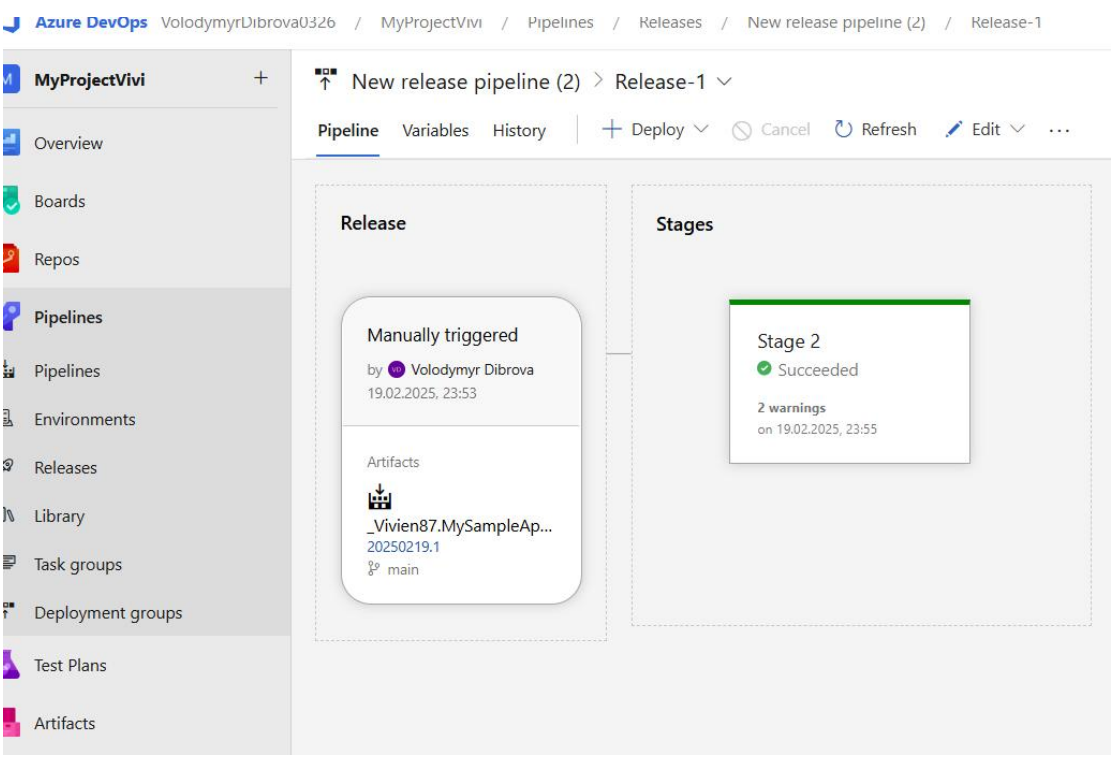
Created a GitHub and pushed from Dev Azure to GitHub

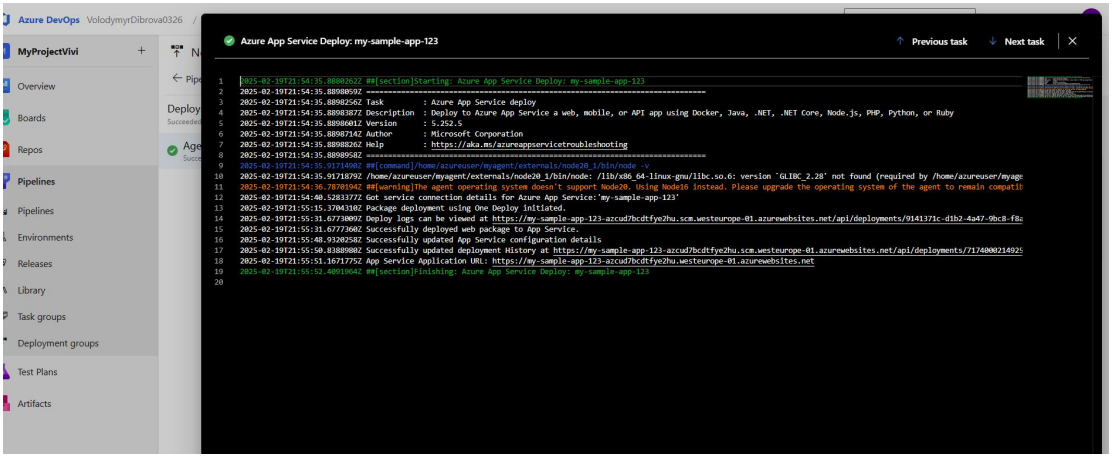



Как думать на SQL?...
Webdesign - Googl...
Настройки
Spring Boot: Excel,...
Vivien87 / MySampleApp
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
MySampleApp
Public
Pin
Unwatch 1
main
1 Branch
0 Tags
Go to file
Add file
Code
Volodymyr Dibrova and Volodymyr Dibrova Updated azure-pipelines.yml
26bd9a9 · 1 hour ago
17 Commits
README.md
Added README.md
yesterday
azure-pipelines.yml
Updated azure-pipelines.yml
1 hour ago
index.js
Updated index.js
yesterday
package.json
Updated package.json
yesterday
README
Introduction
TODO: Give a short introduction of your project. Let this section explain the objectives or the motivation behind this project.

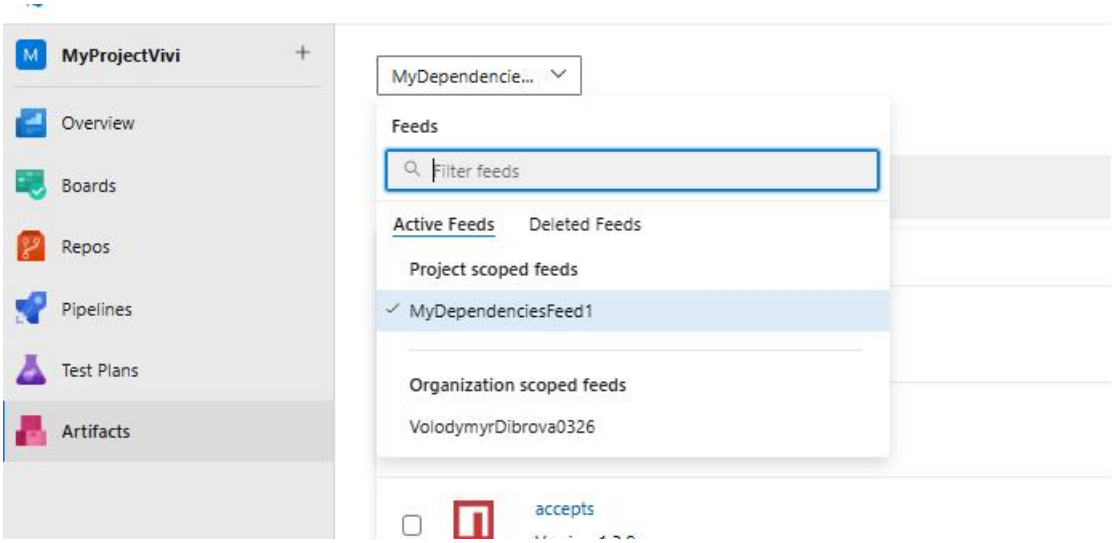

github.com/Vivien87/MySampleApp/blob/main/azure-pipelines.yml
Как думать на SQL?...
Webdesign - Googl...
Настройки
Spring Boot: Excel,...
Files
main
Go to file
README.md
azure-pipelines.yml
index.js
package.json
MySampleApp / azure-pipelines.yml
Vivien87 Update azure-pipelines.yml
Code
Blame
34 lines (26 loc) · 653 Bytes
Code 55% faster with GitHub Copilot
trigger:
- main
pool:
name: MyLinux
demands:
- agent.name -equals myAgent
steps:
- checkout: self
- script: ls -la $(Build.SourcesDirectory)
displayName: 'Checking file in repos'
- task: NodeTool@0
inputs:
versionSpec: '16.x'
displayName: 'Install Node.js'
- script: npm install
displayName: 'Install dependencies'
- script: npm run build
displayName: 'Build the application'

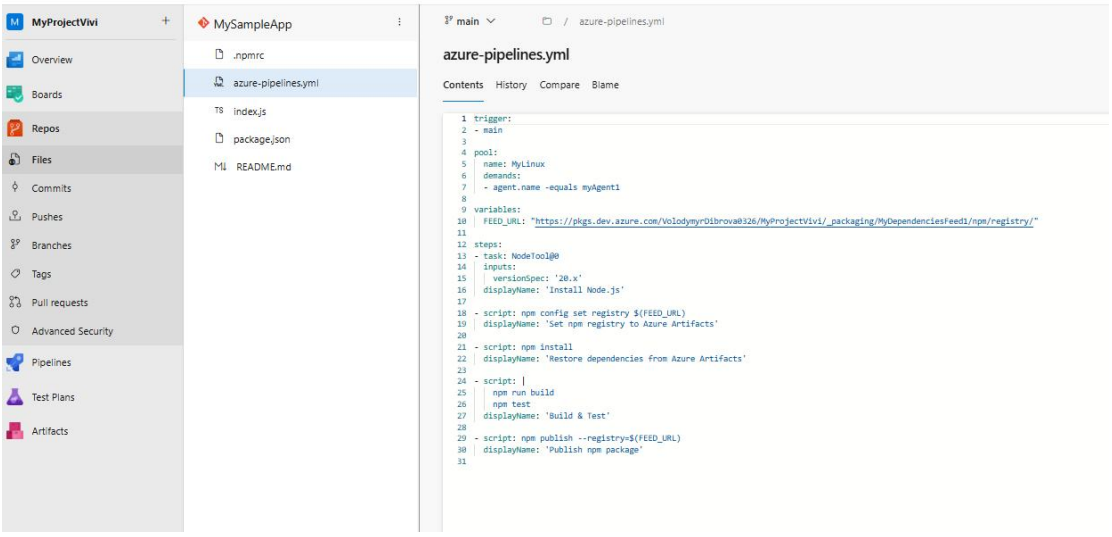

Azure DevOps VolodymyrDibrova0326 / MyProjectVivi / Pipelines / Vivien87.MySampleApp (59) / 20250219.1
Search
MyProjectVivi
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
Library
Task groups
Deployment groups
Test Plans
Artifacts
#20250219.1 • Update azure-pipelines.yml
Vivien87.MySampleApp (59)
Run new
This run is being retained as one of 3 recent runs by pipeline.
View retention leases
Summary
Code Coverage
Manually run by Volodymyr Dibrova
View change
Repository and version
Vivien87/MySampleApp
main
3cdcd073
Time started and elapsed
Just now
37s
Related
0 work items
1 published
Tests and coverage
Get started
Warnings 6
The agent operating system doesn't support Node20. Using Node16 instead. Please upgrade the operating system of the agent to remain compatible with future updates of tasks: https://github.com/no...
Checking file in repos
The agent operating system doesn't support Node20. Using Node16 instead. Please upgrade the operating system of the agent to remain compatible with future updates of tasks: https://github.com/no...
Install Node.js
The agent operating system doesn't support Node20. Using Node16 instead. Please upgrade the operating system of the agent to remain compatible with future updates of tasks: https://github.com/no...
Install dependencies

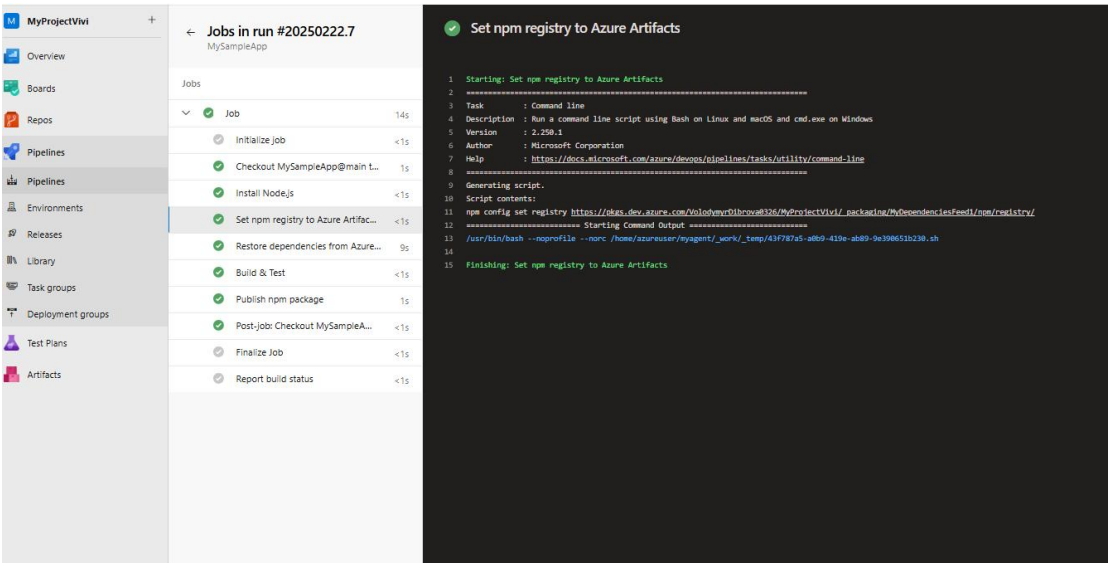

Azure DevOps VolodymyrDibrova0326 / MyProjectVivi / Pipelines / Releases / New release pipeline (2) / Release-1
MyProjectVivi
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
Library
Task groups
Deployment groups
Test Plans
Artifacts
New release pipeline (2) > Release-1
Pipeline
Variables
History
Deploy
Cancel
Refresh
Edit
Release
Manually triggered
by Volodymyr Dibrova
19.02.2025, 23:53
Artifacts
_Vivien87.MySampleAp...
20250219.1
main
Stages
Stage 2
Succeeded
2 warnings
on 19.02.2025, 23:55


Azure App Service Deploy: my-sample-app-123
Previous task
Next task
2025-02-19T21:54:35.8880262Z ##[section]Starting: Azure App Service Deploy: my-sample-app-123
2025-02-19T21:54:35.8898059Z ==========
2025-02-19T21:54:35.8898256Z Task : Azure App Service deploy
2025-02-19T21:54:35.8898387Z Description : Deploy to Azure App Service a web, mobile, or API app using Docker, Java, .NET, .NET Core, Node.js, PHP, Python, or Ruby
2025-02-19T21:54:35.8898601Z Version : 5.252.5
2025-02-19T21:54:35.8898714Z Author : Microsoft Corporation
2025-02-19T21:54:35.8898826Z Help : https://aka.ms/azureappservicetroubleshooting
2025-02-19T21:54:35.8898958Z ==========
2025-02-19T21:54:35.9171490Z ##[command]/home/azureuser/myagent/externals/node20_1/bin/node -v
2025-02-19T21:54:35.9171879Z /home/azureuser/myagent/externals/node20_1/bin/node: /lib/x86_64-linux-gnu/libc.so.6: version `GLIBC_2.28' not found (required by /home/azureuser/myage
2025-02-19T21:54:36.7870194Z ##[warning]The agent operating system doesn't support Node20. Using Node16 instead. Please upgrade the operating system of the agent to remain compatibl
2025-02-19T21:54:40.5283377Z Got service connection details for Azure App Service:'my-sample-app-123'
2025-02-19T21:55:15.3704310Z Package deployment using One Deploy initiated.
2025-02-19T21:55:31.6773009Z Deploy logs can be viewed at https://my-sample-app-123-azcud7bcdtfye2hu.scm.westeurope-01.azurewebsites.net/api/deployments/9141371c-d1b2-4a47-9bc8-f8
2025-02-19T21:55:31.6777360Z Successfully deployed web package to App Service.
2025-02-19T21:55:48.9320258Z Successfully updated App Service configuration details
2025-02-19T21:55:50.8388980Z Successfully updated deployment History at https://my-sample-app-123-azcud7bcdtfye2hu.scm.westeurope-01.azurewebsites.net/api/deployments/7174000214925
2025-02-19T21:55:51.1671775Z App Service Application URL: https://my-sample-app-123-azcud7bcdtfye2hu.westeurope-01.azurewebsites.net
2025-02-19T21:55:52.4091964Z ##[section]Finishing: Azure App Service Deploy: my-sample-app-123

Practical Task 7: Use Azure Artifacts to Manage Dependencies
Requirements:
1. Create an Azure Artifacts Feed:
a. In Azure DevOps, navigate to Artifacts and create a new feed named MyDependenciesFeed.
2. Publish Dependencies:
a. Modify the build pipeline to include a task that publishes the application's
dependencies (e.g., npm packages or NuGet packages) to the Azure Artifacts feed.
b. Use commands like **npm publish** or **dotnet nuget push** in the pipeline.
3. Restore Dependencies:
a. Update the build pipeline to restore dependencies from the Azure Artifacts feed
during the build process.
4. Verify Dependency Management:
a. Run the pipeline and verify that the dependencies are correctly published to and
restored from the feed.

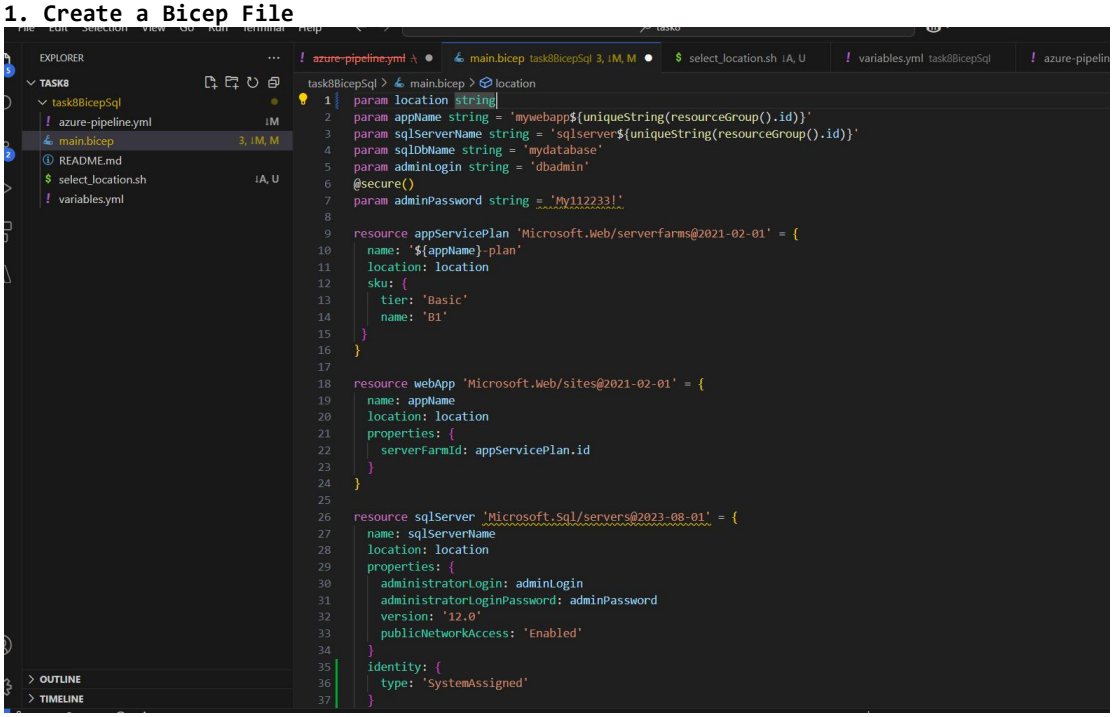






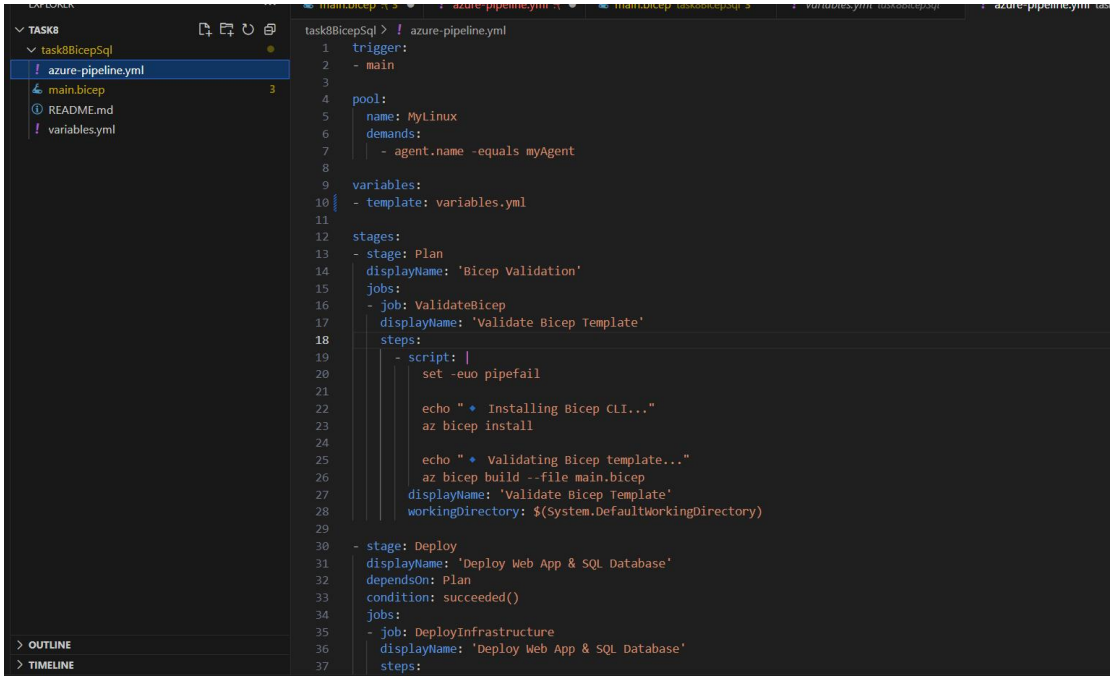
Practical Task 8: Deploy Infrastructure as Code (IaC) with Bicep
Requirements:
1. Create a Bicep File:
a. Create a Bicep file named **main.bicep** that defines the infrastructure for your application, including an Azure Web App and a SQL Database.
2. Create a New Pipeline:
a. Set up a new pipeline in Azure DevOps that includes a stage for deploying the Bicep
file.
3. Deploy Using Azure CLI:
a. Use the Azure CLI task in the pipeline to deploy the Bicep file.
b. The command should look like:***az deployment group create --resource-group <YourResourceGroup> --template-file main.bicep***
4. Run the Pipeline:
a. Save and run the pipeline to ensure that the infrastructure is provisioned correctly in
Azure.
b. Verify that the resources are created in the Azure portal.

1. Create a Bicep File

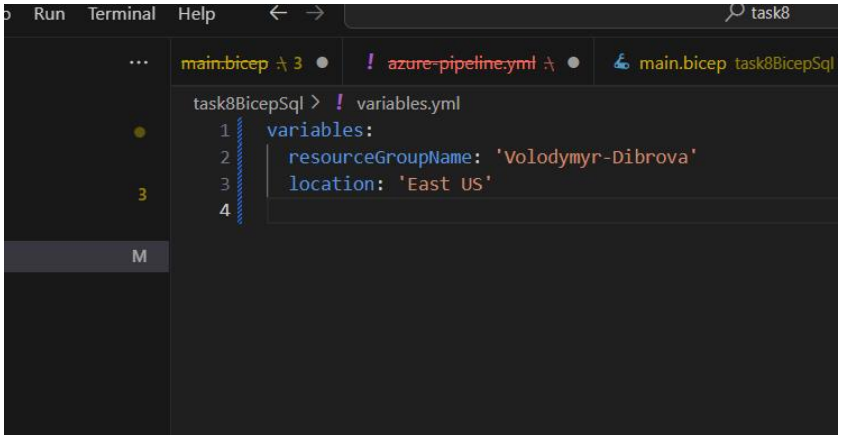


```
param location string
param appName string = 'mywebapp${uniqueString(resourceGroup().id)}'
param sqlServerName string = 'sqlserver${uniqueString(resourceGroup().id)}'
param sqlDbName string = 'mydatabase'
param adminLogin string = 'dbadmin'
@secure()
param adminPassword string = 'My112233!'

resource appServicePlan 'Microsoft.Web/serverfarms@2021-02-01' = {
  name: '${appName}-plan'
  location: location
  sku: {
    tier: 'Basic'
    name: 'B1'
  }
}

resource webApp 'Microsoft.Web/sites@2021-02-01' = {
  name: appName
  location: location
  properties: {
    serverFarmId: appServicePlan.id
  }
}

resource sqlServer 'Microsoft.Sql/servers@2023-08-01' = {
  name: sqlServerName
  location: location
  properties: {
    administratorLogin: adminLogin
    administratorLoginPassword: adminPassword
    version: '12.0'
    publicNetworkAccess: 'Enabled'
  }
  identity: {
    type: 'SystemAssigned'
  }
```

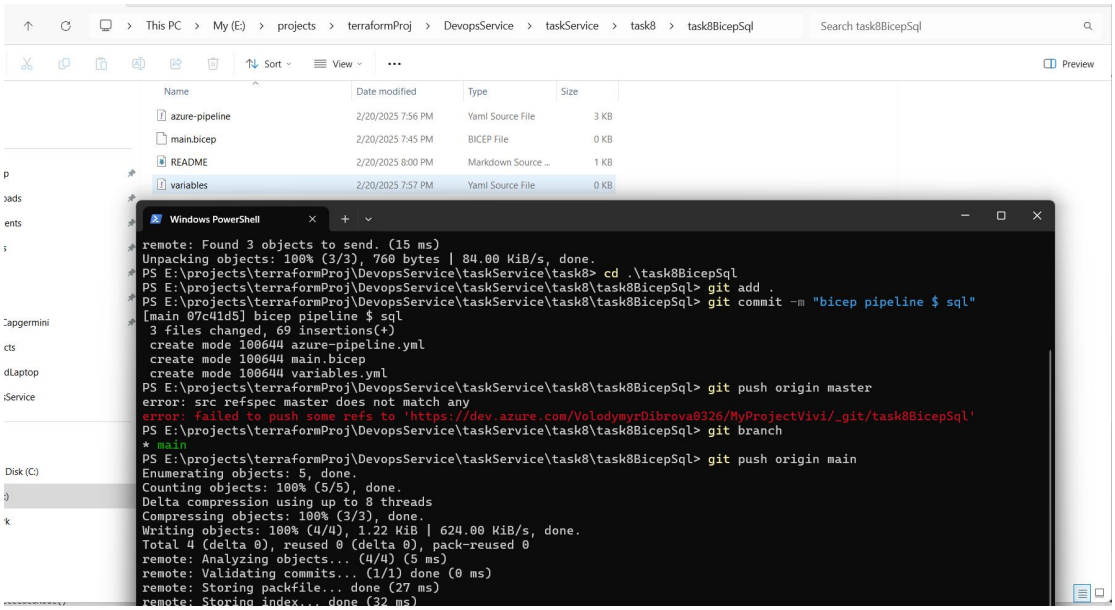
Created pipeline



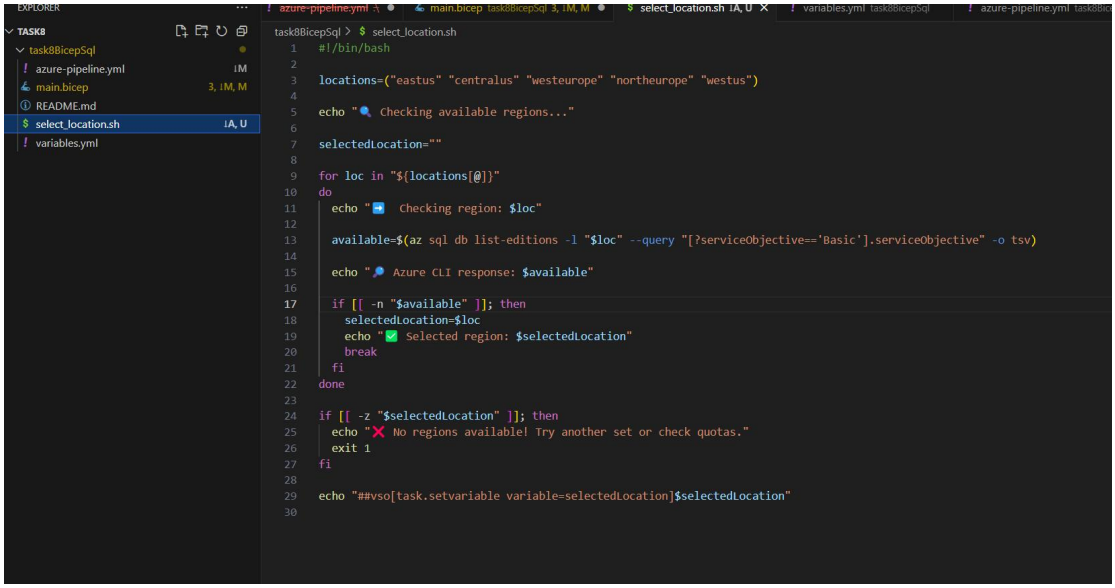
```
trigger:
- main

pool:
  name: MyLinux
  demands:
    - agent.name -equals myAgent

variables:
- template: variables.yml

stages:
- stage: Plan
  displayName: 'Bicep Validation'
  jobs:
  - job: ValidateBicep
    displayName: 'Validate Bicep Template'
    steps:
      - script: |
          set -euo pipefail

          echo "🔹 Installing Bicep CLI..."
          az bicep install

          echo "🔹 Validating Bicep template..."
          az bicep build --file main.bicep
        displayName: 'Validate Bicep Template'
        workingDirectory: $(System.DefaultWorkingDirectory)

- stage: Deploy
  displayName: 'Deploy Web App & SQL Database'
  dependsOn: Plan
  condition: succeeded()
  jobs:
  - job: DeployInfrastructure
    displayName: 'Deploy Web App & SQL Database'
    steps:
```

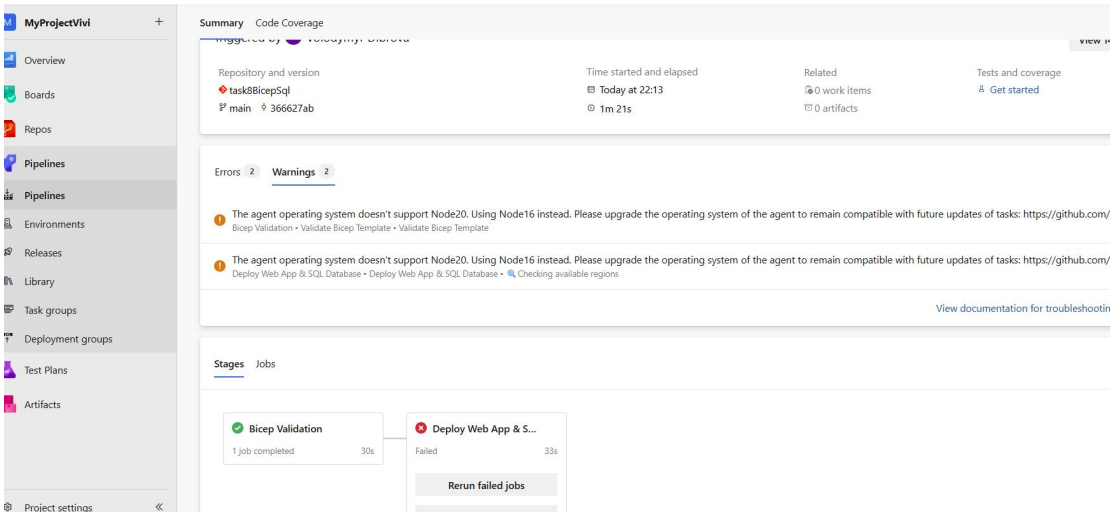


Pushed to Repo



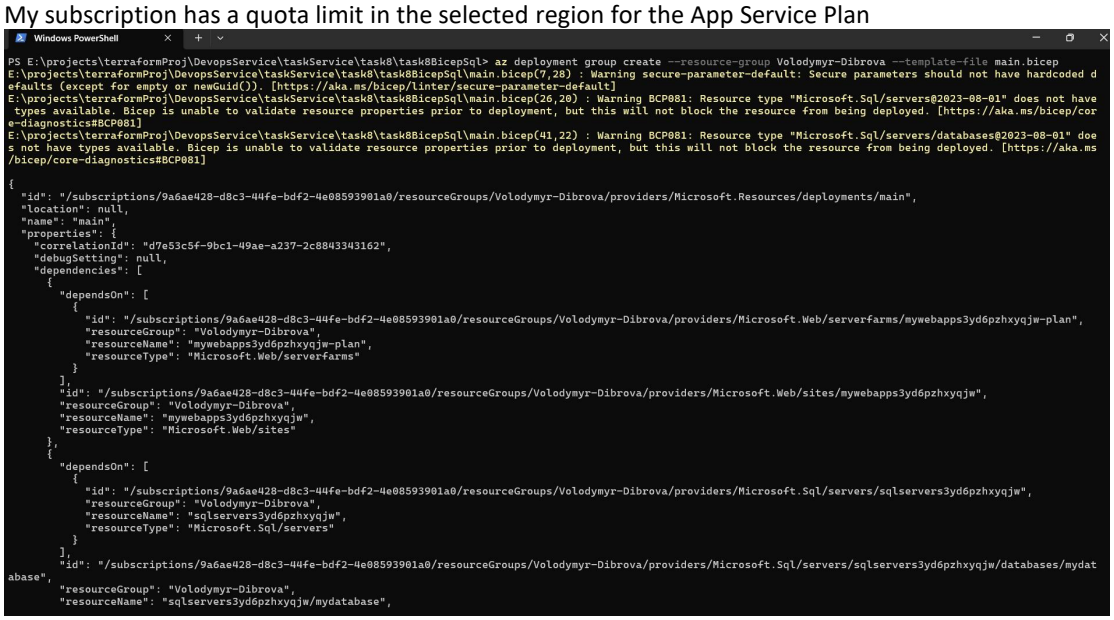
Created a program to search for available regions

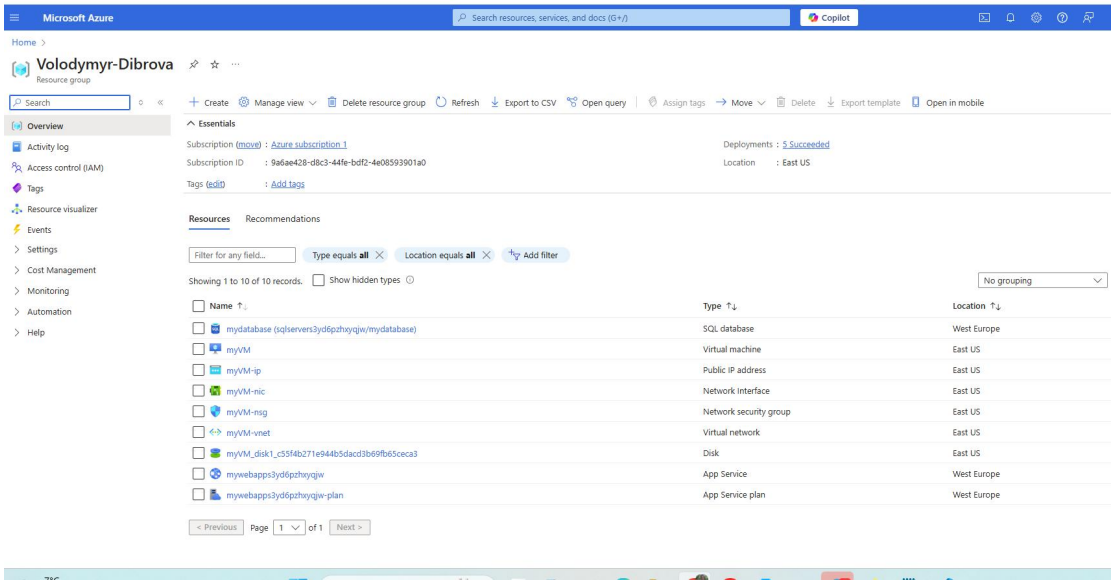




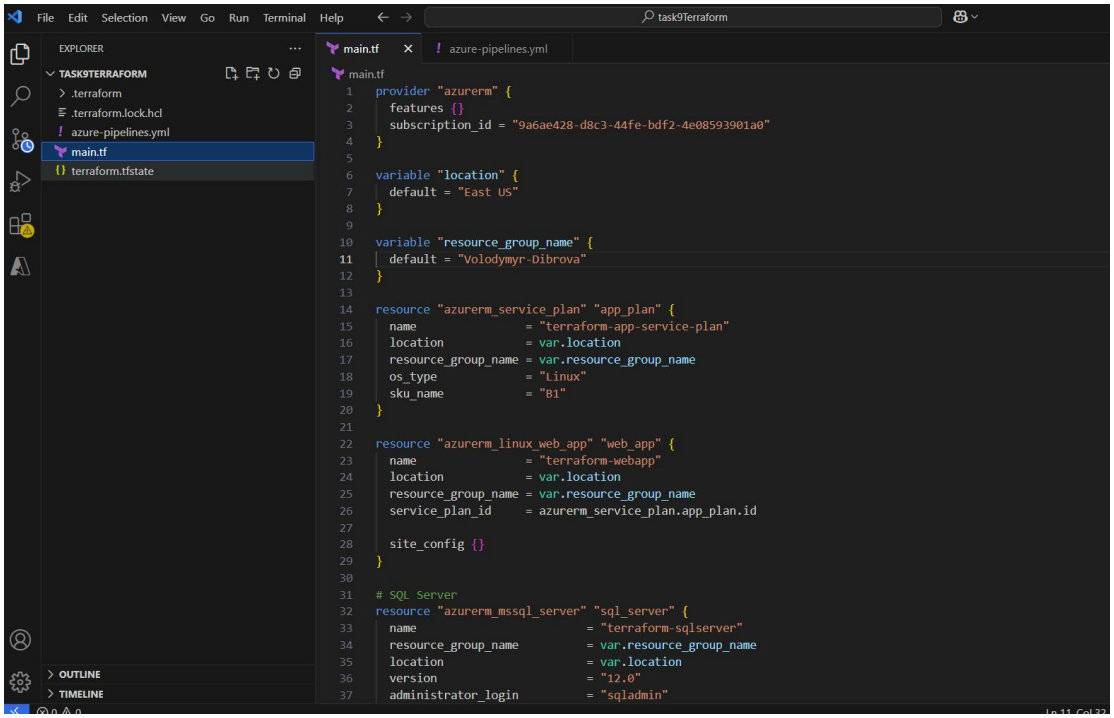

```
2025-02-20T20:15:01.6339276Z
2025-02-20T20:15:01.6339634Z Python (Linux) 3.11.8 (main, Apr 24 2024, 04:14:33) [GCC 7.5.0]
2025-02-20T20:15:01.6339790Z
2025-02-20T20:15:01.6340100Z Legal docs and information: aka.ms/AzureCliLegal
2025-02-20T20:15:01.6340275Z
2025-02-20T20:15:01.6340362Z
2025-02-20T20:15:01.6340689Z Setting AZURE_CONFIG_DIR env variable to: /home/azureuser/myagent/_work/_temp/.azclitask
2025-02-20T20:15:01.6341091Z Setting active cloud to: AzureCloud
2025-02-20T20:15:01.6341444Z [command]/usr/bin/az cloud set -n AzureCloud
2025-02-20T20:15:04.8235532Z [command]/usr/bin/az login --service-principal -u *** --tenant 8d1157bb-1f96-415f-824b-ab0a29485d7d --allow-no-subscriptions --federated-token ***
2025-02-20T20:15:06.2435532Z [
2025-02-20T20:15:06.2436239Z   {
2025-02-20T20:15:06.2436595Z     "cloudName": "AzureCloud",
2025-02-20T20:15:06.2437010Z     "homeTenantId": "8d1157bb-1f96-415f-824b-ab0a29485d7d",
2025-02-20T20:15:06.2437439Z     "id": "9a6ae428-d8c3-44fe-bdf2-4e08593901a0",
2025-02-20T20:15:06.2437811Z     "isDefault": true,
2025-02-20T20:15:06.2438155Z     "managedByTenants": [],
2025-02-20T20:15:06.2438511Z     "name": "Azure subscription 1",
2025-02-20T20:15:06.2438861Z     "state": "Enabled",
2025-02-20T20:15:06.2439224Z     "tenantId": "8d1157bb-1f96-415f-824b-ab0a29485d7d",
2025-02-20T20:15:06.2439571Z     "user": {
2025-02-20T20:15:06.2440070Z       "name": "***",
2025-02-20T20:15:06.2440422Z       "type": "servicePrincipal"
2025-02-20T20:15:06.2440743Z     }
2025-02-20T20:15:06.2441038Z   }
2025-02-20T20:15:06.2441338Z ]
2025-02-20T20:15:06.2441716Z [command]/usr/bin/az account set --subscription 9a6ae428-d8c3-44fe-bdf2-4e08593901a0
2025-02-20T20:15:06.7915138Z [command]/bin/bash /home/azureuser/myagent/_work/_temp/azureclitaskscript1740082501016.sh
2025-02-20T20:15:06.8804019Z 🔍 Checking available regions...
2025-02-20T20:15:06.8822086Z ➡️  Checking region: eastus
2025-02-20T20:15:08.2479200Z 🔑 Azure CLI response:
2025-02-20T20:15:08.2480329Z ➡️  Checking region: centralus
2025-02-20T20:15:09.5460214Z 🔑 Azure CLI response:
2025-02-20T20:15:09.5461231Z ➡️  Checking region: westeurope
2025-02-20T20:15:11.3728467Z 🔑 Azure CLI response:
2025-02-20T20:15:11.3729537Z ➡️  Checking region: northeurope
2025-02-20T20:15:13.9699956Z 🔑 Azure CLI response:
2025-02-20T20:15:13.9701474Z ➡️  Checking region: westus
2025-02-20T20:15:15.4235815Z 🔑 Azure CLI response:
2025-02-20T20:15:15.4236959Z ❌ No regions available! Try another set or check quotas.
2025-02-20T20:15:15.4269080Z ##[error]Script failed with exit code: 1
2025-02-20T20:15:15.4289804Z [command]/usr/bin/az account clear
2025-02-20T20:15:16.3368236Z ##[section]Finishing: 🔍Checking available regions
```

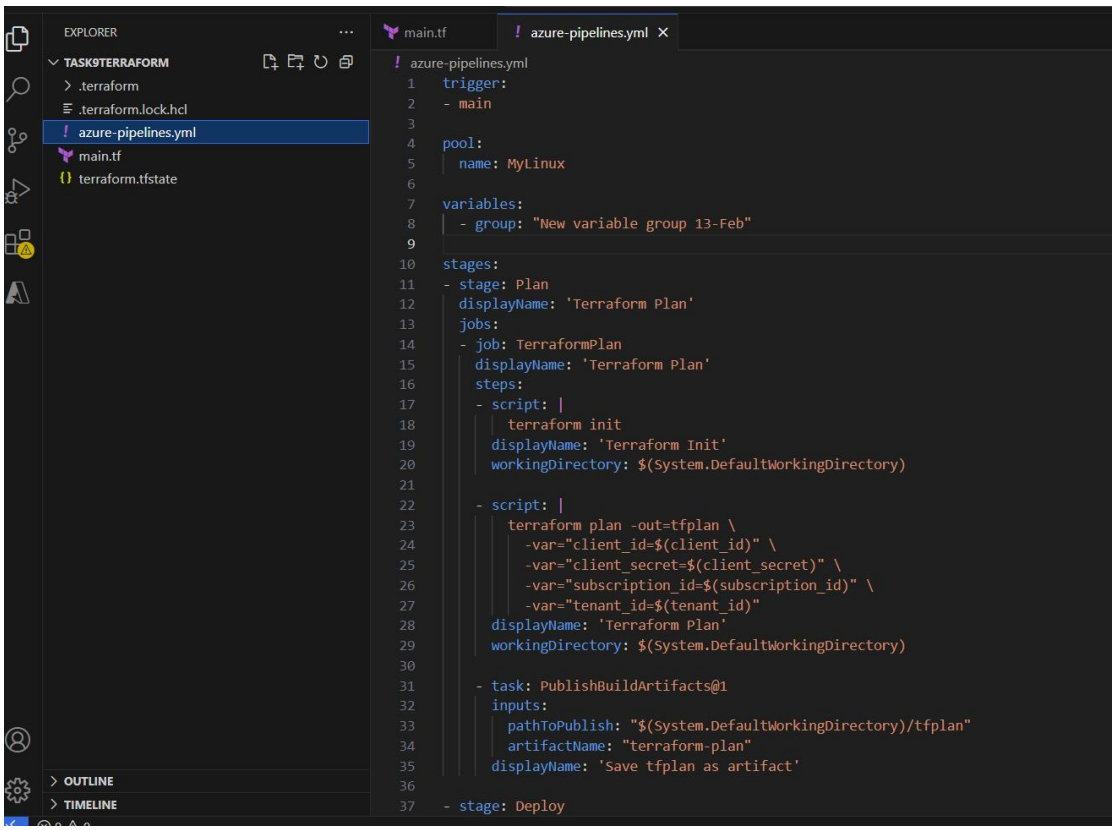
My subscription has a quota limit in the selected region for the App Service Plan





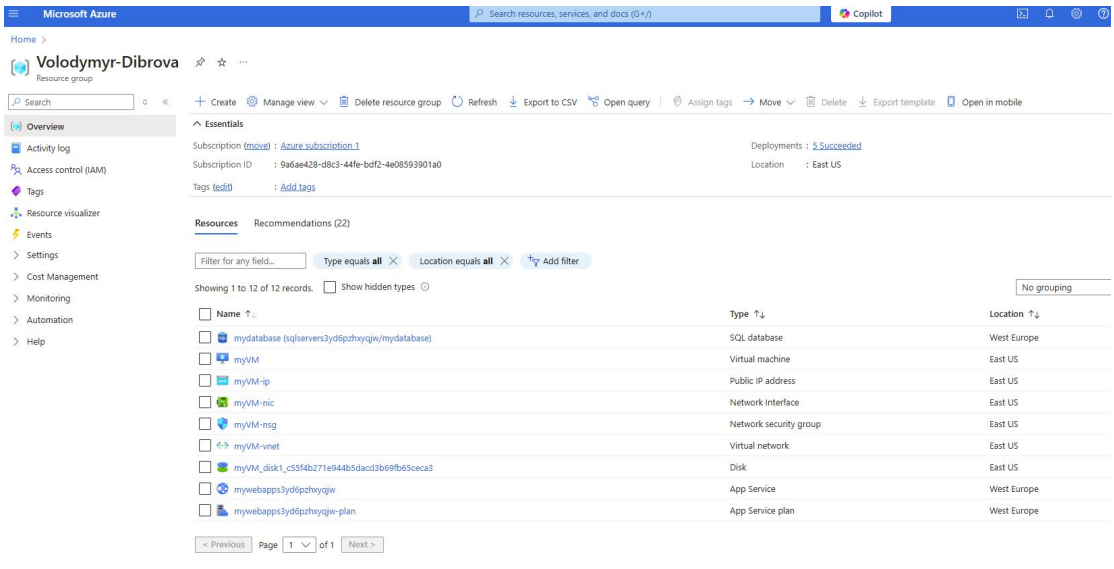
Practical Task 9: Implement Infrastructure as Code (IaC) with Terraform
Requirements:
1. Create a Terraform Configuration:
a. Create a Terraform configuration file (main.tf) that defines the same infrastructure
as in Task 8 (e.g., Azure Web App and SQL Database).
2. Set Up a New Azure Pipeline:
a. Create a new Azure Pipeline that runs Terraform commands.
b. Use the following steps in the pipeline:
terraform init to initialize the Terraform working directory.
terraform plan to create an execution plan.
terraform apply -auto-approve to apply the changes.
3. Configure Triggers:
a. Set up the pipeline to trigger on changes to the Terraform files in the repository.
4. Run the Pipeline:
a. Save and run the pipeline to provision the infrastructure in Azure.
b. Verify that the resources are created as expected.

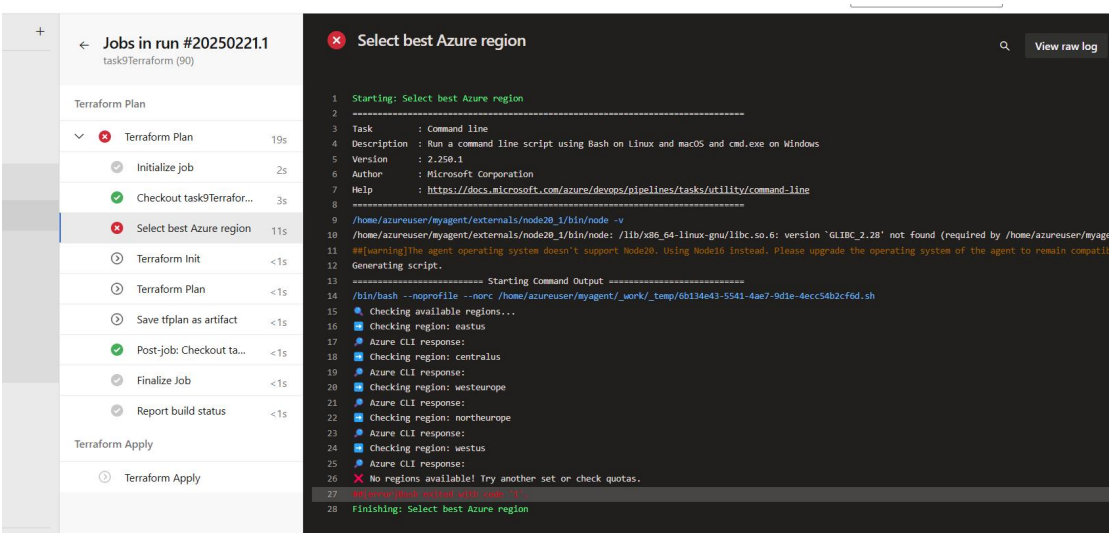

File Edit Selection View Go Run Terminal Help
task9Terraform
EXPLORER
TASK9TERRAFORM
.terraform
.terraform.lock.hcl
azure-pipelines.yml
main.tf
terraform.tfstate
main.tf
azure-pipelines.yml
OUTLINE
TIMELINE
provider "azurerm" {
features {}
subscription_id = "9a6ae428-d8c3-44fe-bdf2-4e08593901a0"
}
variable "location" {
default = "East US"
}
variable "resource_group_name" {
default = "Volodymyr-Dibrova"
}
resource "azurerm_service_plan" "app_plan" {
name = "terraform-app-service-plan"
location = var.location
resource_group_name = var.resource_group_name
os_type = "Linux"
sku_name = "B1"
}
resource "azurerm_linux_web_app" "web_app" {
name = "terraform-webapp"
location = var.location
resource_group_name = var.resource_group_name
service_plan_id = azurerm_service_plan.app_plan.id
site_config {}
}
# SQL Server
resource "azurerm_mssql_server" "sql_server" {
name = "terraform-sqlserver"
resource_group_name = var.resource_group_name
location = var.location
version = "12.0"
administrator_login = "sqladmin"


EXPLORER
TASK9TERRAFORM
.terraform
.terraform.lock.hcl
azure-pipelines.yml
main.tf
terraform.tfstate
main.tf
azure-pipelines.yml
OUTLINE
TIMELINE
trigger:
- main
pool:
name: MyLinux
variables:
- group: "New variable group 13-Feb"
stages:
- stage: Plan
displayName: 'Terraform Plan'
jobs:
- job: TerraformPlan
displayName: 'Terraform Plan'
steps:
- script: |
terraform init
displayName: 'Terraform Init'
workingDirectory: $(System.DefaultWorkingDirectory)
- script: |
terraform plan -out=tfplan \
-var="client_id=$(client_id)" \
-var="client_secret=$(client_secret)" \
-var="subscription_id=$(subscription_id)" \
-var="tenant_id=$(tenant_id)"
displayName: 'Terraform Plan'
workingDirectory: $(System.DefaultWorkingDirectory)
- task: PublishBuildArtifacts@1
inputs:
pathToPublish: "$(System.DefaultWorkingDirectory)/tfplan"
artifactName: "terraform-plan"
displayName: 'Save tfplan as artifact'
- stage: Deploy

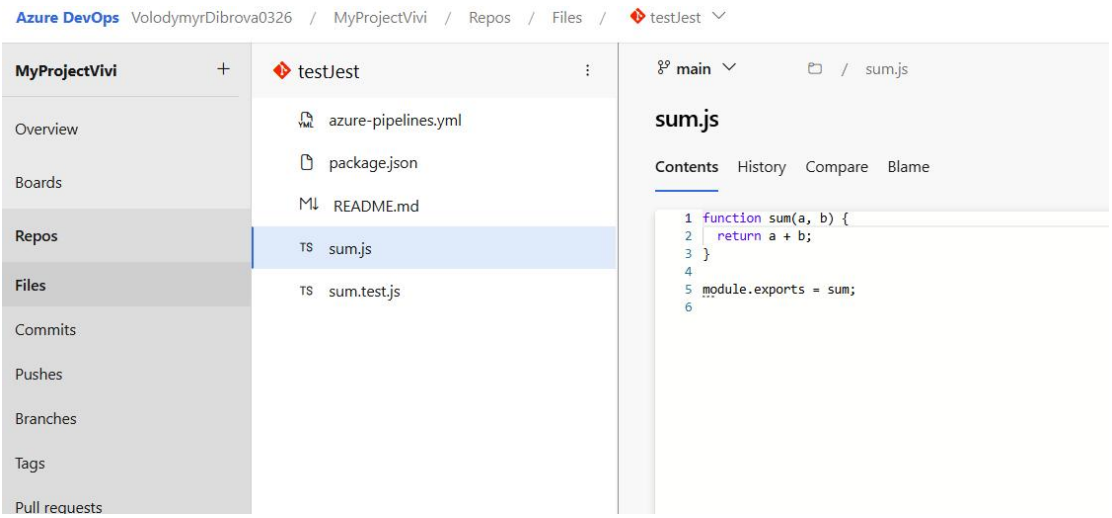
```
you run "terraform apply" now.
PS E:\projects\terraformProj\DevopsService\taskService\task9Terraform> terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
  + create

Terraform will perform the following actions:

  # azurerm_linux_web_app.web_app will be created
  + resource "azurerm_linux_web_app" "web_app" {
      + client_affinity_enabled                        = false
      + client_certificate_enabled                     = false
      + client_certificate_mode                        = "Required"
      + custom_domain_verification_id                  = (sensitive value)
      + default_hostname                               = (known after apply)
      + enabled                                        = true
      + ftp_publish_basic_authentication_enabled       = true
      + hosting_environment_id                         = (known after apply)
      + https_only                                     = false
      + id                                             = (known after apply)
      + key_vault_reference_identity_id                = (known after apply)
      + kind                                           = (known after apply)
      + location                                       = "eastus"
      + name                                           = "terraform-webapp"
      + outbound_ip_address_list                       = (known after apply)
      + outbound_ip_addresses                          = (known after apply)
      + possible_outbound_ip_address_list              = (known after apply)
      + possible_outbound_ip_addresses                 = (known after apply)
      + public_network_access_enabled                  = true
      + resource_group_name                            = "Volodymyr-Dibrova"
      + service_plan_id                                = (known after apply)
      + site_credential                                = (sensitive value)
      + webdeploy_publish_basic_authentication_enabled = true
      + zip_deploy_file                                = (known after apply)

      + site_config {
          + always_on                                   = true
          + container_registry_use_managed_identity = false
          + default_documents                           = (known after apply)
```

```
azurerm_mssql_database.sql_db: Still creating... [1m50s elapsed]
azurerm_mssql_database.sql_db: Still creating... [2m0s elapsed]
azurerm_mssql_database.sql_db: Still creating... [2m10s elapsed]
azurerm_mssql_database.sql_db: Still creating... [2m20s elapsed]
azurerm_mssql_database.sql_db: Creation complete after 2m27s [id=/subscriptions/9a6ae428-d8c3-44fe-bdf2-4e08593901a0/resourceGroups/Volodymyr-Dibrova/providers/Microsoft.Sql/servers/terraform-sqlserver/databases/terraform-db]

╷
│ Error: creating App Service Plan (Subscription: "9a6ae428-d8c3-44fe-bdf2-4e08593901a0"
│ Resource Group Name: "Volodymyr-Dibrova"
│ Server Farm Name: "terraform-app-service-plan"): performing CreateOrUpdate: unexpected status 401 (401 Unauthorized) with response: {"Code":"Unauthorized"
,"Message":"This region has quota of 0 instances for your subscription. Try selecting different region or SKU.","Target":null,"Details":[{"Message":"This re
gion has quota of 0 instances for your subscription. Try selecting different region or SKU."},{"Code":"Unauthorized"},{"ErrorEntity":{"ExtendedCode":"52039"
,"MessageTemplate":"{0}. Try selecting different region or SKU.","Parameters":["This region has quota of 0 instances for your subscription"],"Code":"Unautho
rized","Message":"This region has quota of 0 instances for your subscription. Try selecting different region or SKU."}}],"Innererror":null}
│
│   with azurerm_service_plan.app_plan,
│   on main.tf line 14, in resource "azurerm_service_plan" "app_plan":
│   14: resource "azurerm_service_plan" "app_plan" {
│
│ creating App Service Plan (Subscription: "9a6ae428-d8c3-44fe-bdf2-4e08593901a0"
│ Resource Group Name: "Volodymyr-Dibrova"
│ Server Farm Name: "terraform-app-service-plan"): performing CreateOrUpdate: unexpected status 401 (401 Unauthorized)
│ with response: {"Code":"Unauthorized","Message":"This region has quota of 0 instances for your subscription. Try
│ selecting different region or SKU.","Target":null,"Details":[{"Message":"This region has quota of 0 instances for
│ your subscription. Try selecting different region or
│ SKU."},{"Code":"Unauthorized"},{"ErrorEntity":{"ExtendedCode":"52039","MessageTemplate":"{0}. Try selecting different
│ region or SKU.","Parameters":["This region has quota of 0 instances for your
│ subscription"],"Code":"Unauthorized","Message":"This region has quota of 0 instances for your subscription. Try
│ selecting different region or SKU."}}],"Innererror":null}

PS E:\projects\terraformProj\DevopsService\taskService\task9Terraform>
```

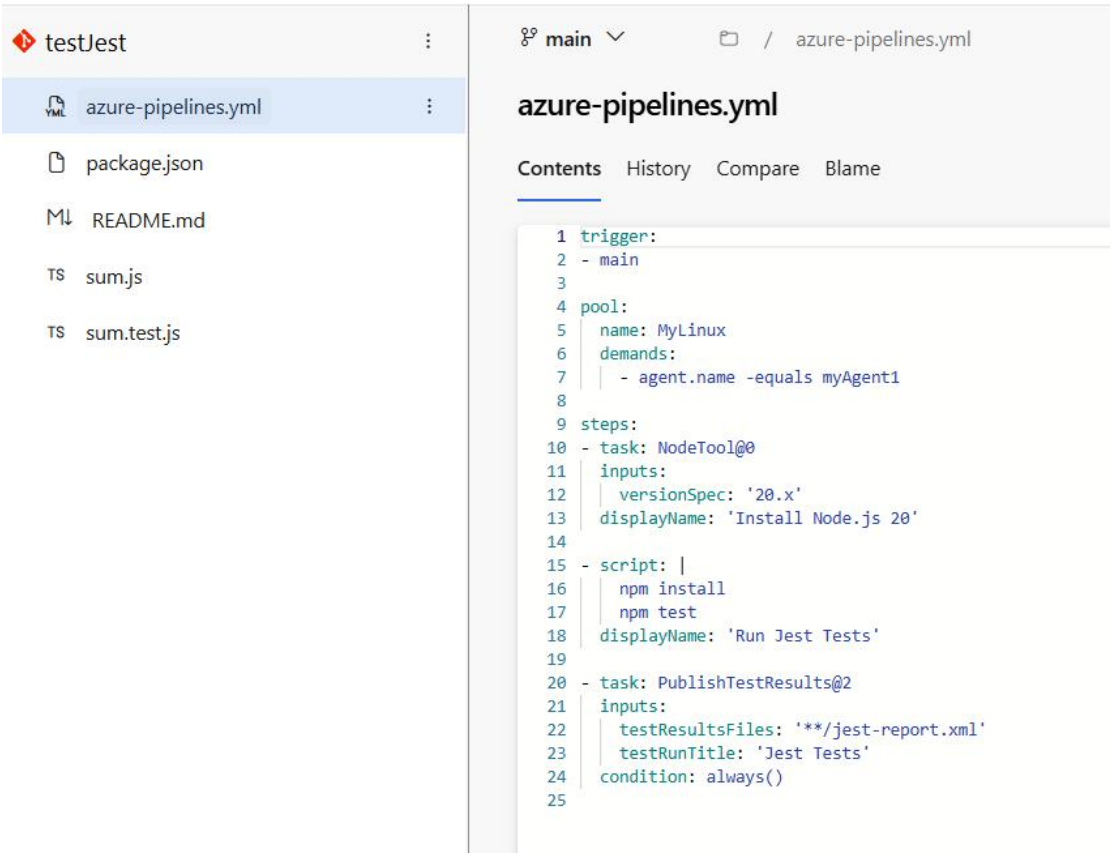


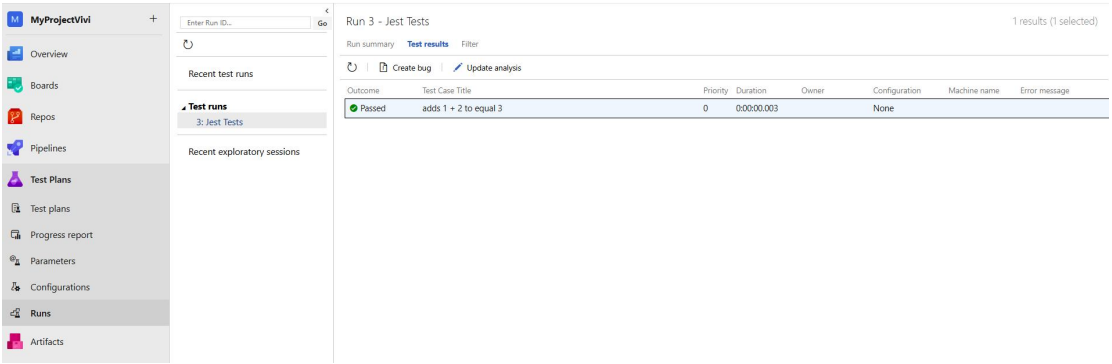


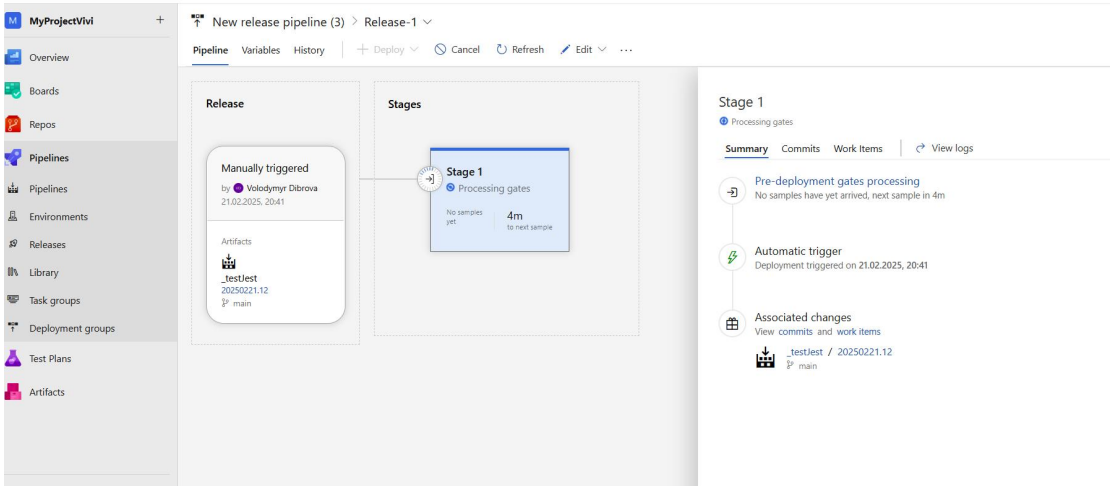
Practical Task 10: Integrate Azure Test Plans with CI/CD Pipelines
Requirements:
1. Create a Test Plan:
a. Navigate to Azure Test Plans and create a new test plan that includes both manual
and automated tests for your application.
2. Modify the Build Pipeline:
a. Update the build pipeline to include a testing stage that runs automated tests using
a testing framework (e.g., NUnit for .NET or Jest for Node.js).
b. Add a task in the pipeline to publish the test results.
3. Configure Release Pipeline:
a. Modify the release pipeline to deploy the application only if all tests pass.
b. Use conditions in the release pipeline to check for successful test results.
4. Run the Pipeline:a. Trigger the pipeline and verify that it runs the tests and reports the results back to
Azure Test Plans.
b. Ensure that any failed tests block the deployment process.

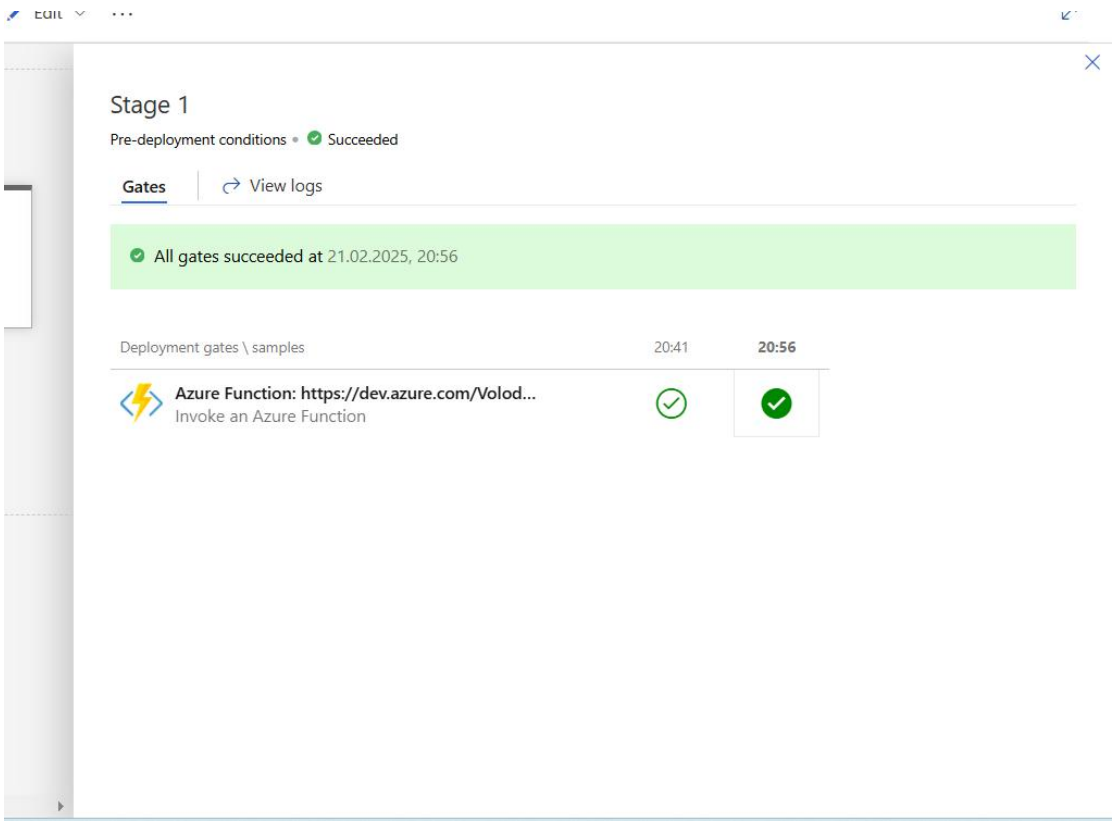



azure-pipelines.yml
package.json
README.md
sum.js
sum.test.js
sum.test.js
Contents History Compare Blame
1 const sum = require("./sum");
2
3 test("adds 1 + 2 to equal 3", () => {
4 expect(sum(1, 2)).toBe(3);
5 });
6


testJest
azure-pipelines.yml
package.json
README.md
sum.js
sum.test.js
main / azure-pipelines.yml
azure-pipelines.yml
Contents History Compare Blame
1 trigger:
2 - main
3
4 pool:
5 name: MyLinux
6 demands:
7 - agent.name -equals myAgent1
8
9 steps:
10 - task: NodeTool@0
11 inputs:
12 versionSpec: '20.x'
13 displayName: 'Install Node.js 20'
14
15 - script: |
16 npm install
17 npm test
18 displayName: 'Run Jest Tests'
19
20 - task: PublishTestResults@2
21 inputs:
22 testResultsFiles: '**/jest-report.xml'
23 testRunTitle: 'Jest Tests'
24 condition: always()
25


MyProjectVivi
Overview
Boards
Repos
Pipelines
Test Plans
Test plans
Progress report
Parameters
Configurations
Runs
Artifacts
Recent test runs
Test runs
3: Jest Tests
Recent exploratory sessions
Run 3 - Jest Tests
1 results (1 selected)
Run summary
Test results
Filter
Create bug
Update analysis
Outcome
Test Case Title
Priority
Duration
Owner
Configuration
Machine name
Error message
Passed
adds 1 + 2 to equal 3
0
0:00:00.003
None





Practical Task 11: Full CI/CD Pipeline with Infrastructure as Code, Testing, and Artifact Management

Requirements:

1. Design the Application Architecture:

a. Create a simple application that consists of multiple components (e.g., a frontend application, a backend API, and a database).

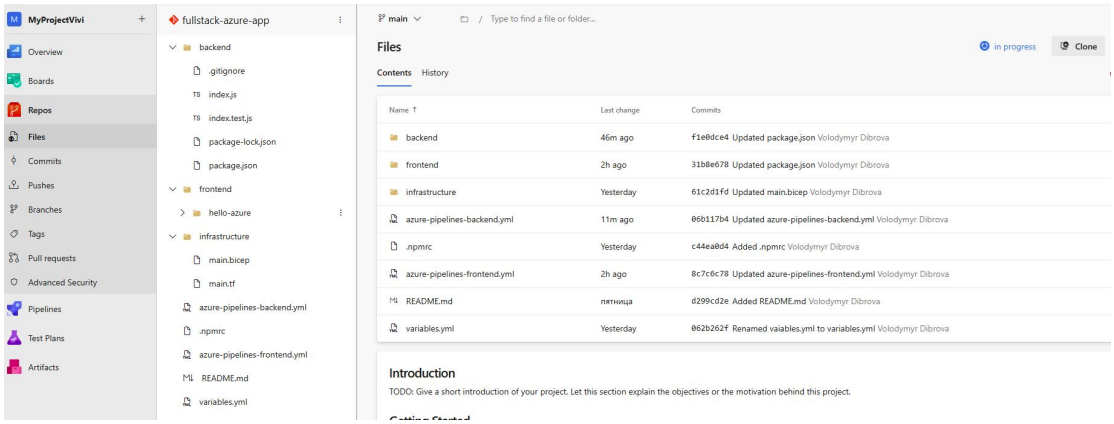
2. Create a Bicep File for Infrastructure:

a. Create a Bicep file (main.bicep) that provisions all necessary resources for the application, including:

i. An Azure App Service for the frontend.

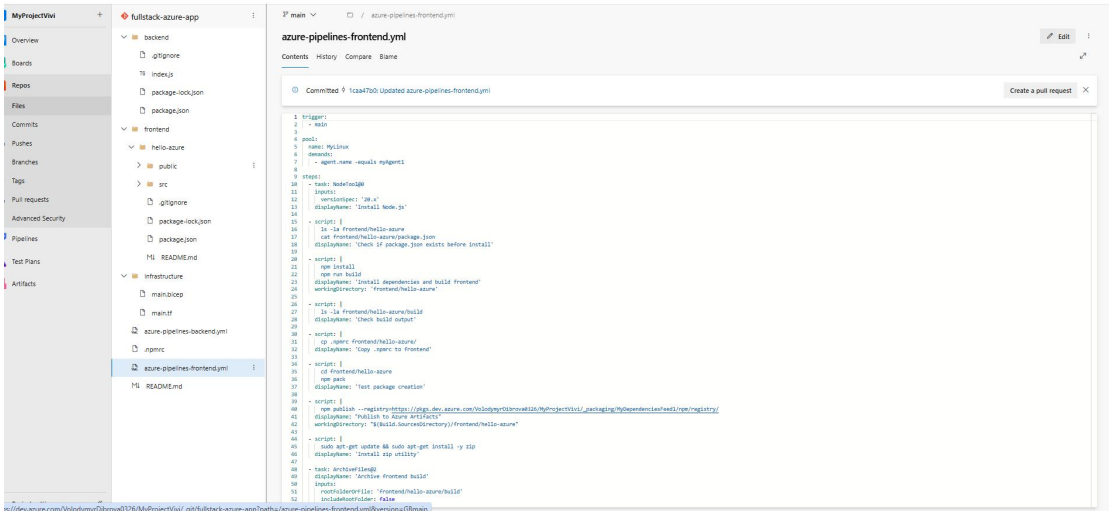
ii. An Azure Function or Azure App Service for the backend API.
iii. An Azure SQL Database or Cosmos DB for data storage.
b. Ensure that the Bicep file includes networking components, such as a Virtual
Network and Subnets if needed.
3. Create Terraform Configuration:
a. Create a Terraform configuration (main.tf) that defines the same infrastructure as
the Bicep file.
b. Ensure that the Terraform configuration can be used to provision the infrastructure
independently.
4. Set Up Azure Artifacts:
a. Create an Azure Artifacts feed to store the application's packages (e.g., npm
packages for the frontend and NuGet packages for the backend).
b. Modify the build pipelines to publish the application's packages to the Azure
Artifacts feed.
5. Create CI/CD Pipelines:
a. Create two separate pipelines: one for the frontend and one for the backend.
6. Frontend Pipeline:
a. The pipeline should:
i. Trigger on changes to the frontend code in the repository.
ii. Build the frontend application and run unit tests.
iii. Publish the build artifacts to Azure Artifacts.
iv. Deploy the frontend application to the Azure App Service using the Bicep
file.
7. Backend Pipeline:
a. The pipeline should:
i. Trigger on changes to the backend code in the repository.
ii. Build the backend API and run unit tests.
iii. Publish the backend API package to Azure Artifacts.iv. Deploy the backend API to Azure using the Bicep file.
8. Integrate Azure Test Plans:
a. Create a comprehensive test plan in Azure Test Plans that includes both manual and
automated tests for both the frontend and backend.
b. Update both pipelines to include a testing stage that runs automated tests and
publishes the results to Azure Test Plans.
9. Implement Continuous Monitoring:
a. Set up Application Insights for both the frontend and backend applications to
monitor performance and errors.
b. Configure alerts for critical issues that may arise in the production environment.
10. Run the Full CI/CD Process:
a. Trigger both pipelines by making changes to the codebase.
b. Verify that the entire CI/CD process runs smoothly, from code commit to deployment.
c. Check the Azure portal to ensure all resources are provisioned correctly and that the
applications are running as expected.

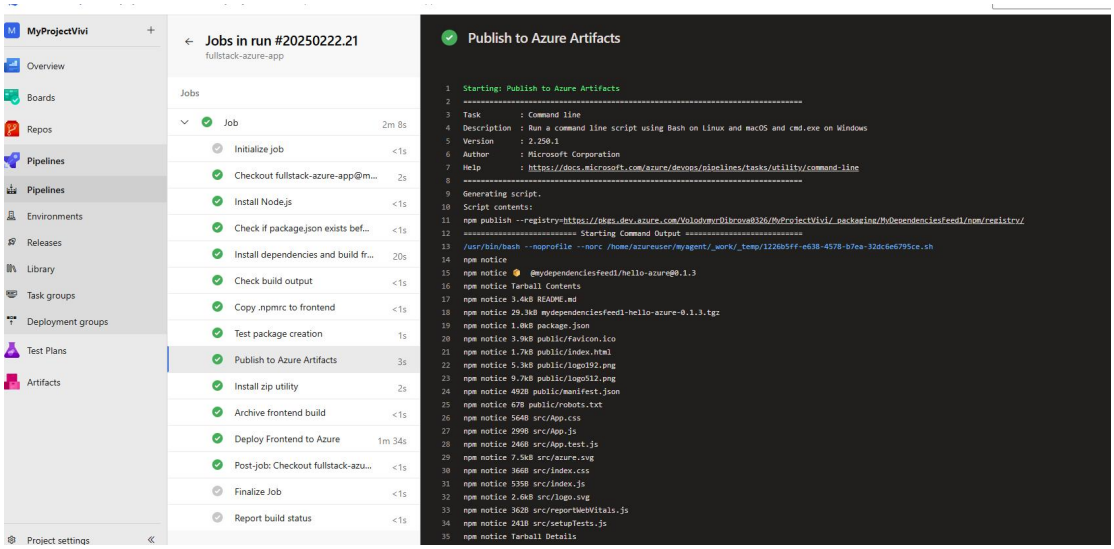
My project structure

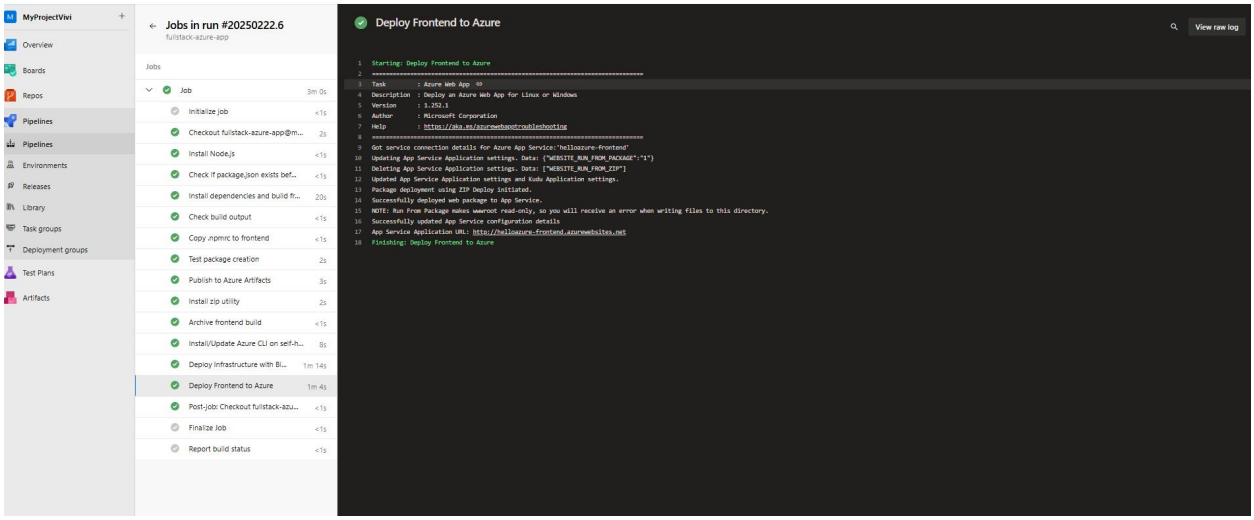


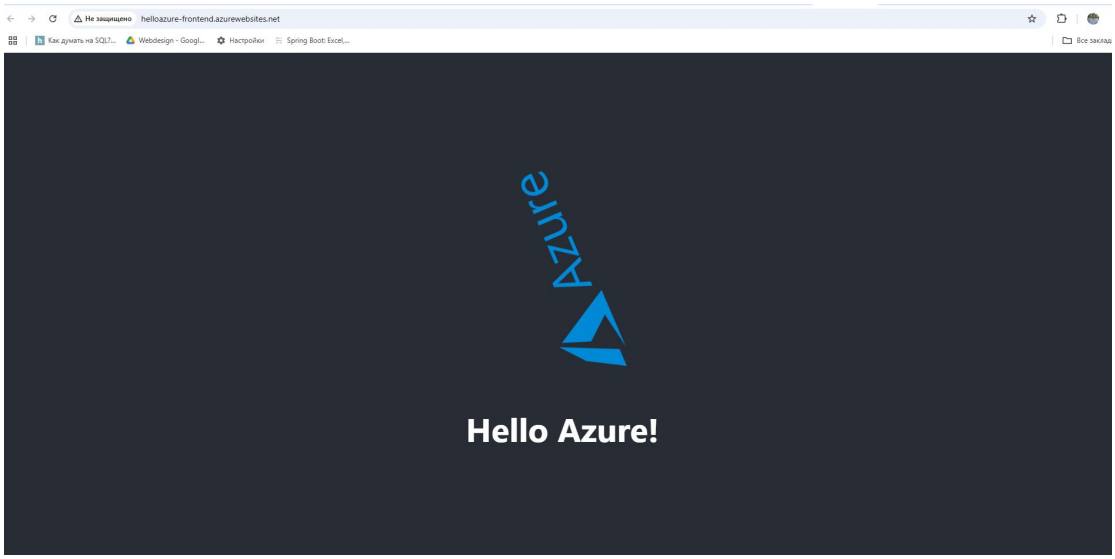
1.Created a simple application that consists of multiple components a frontend application, a backend API, and a database

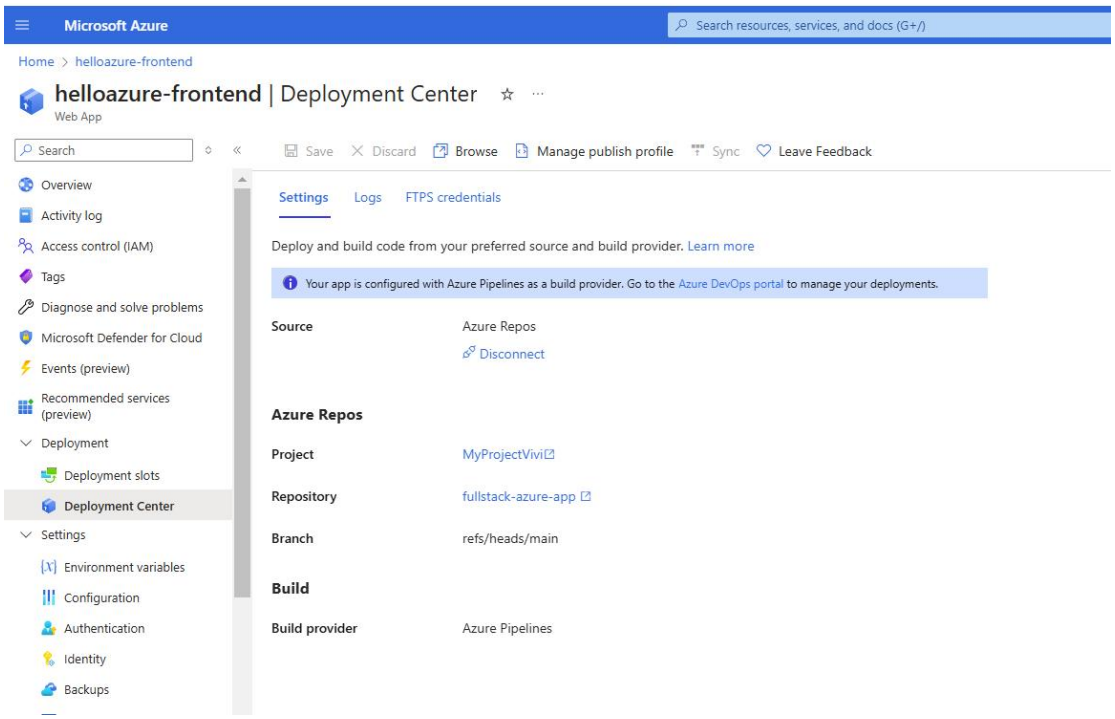
frontend



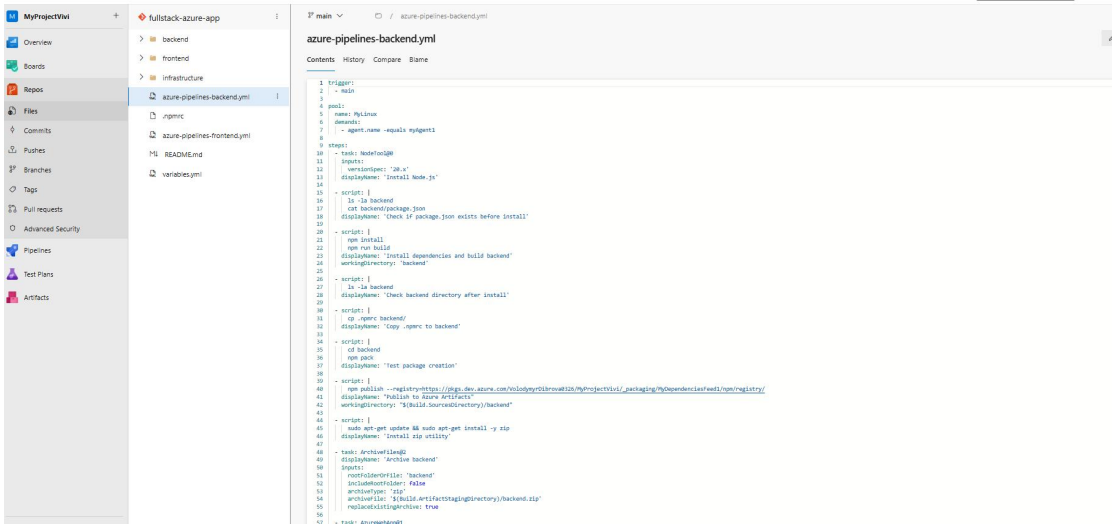


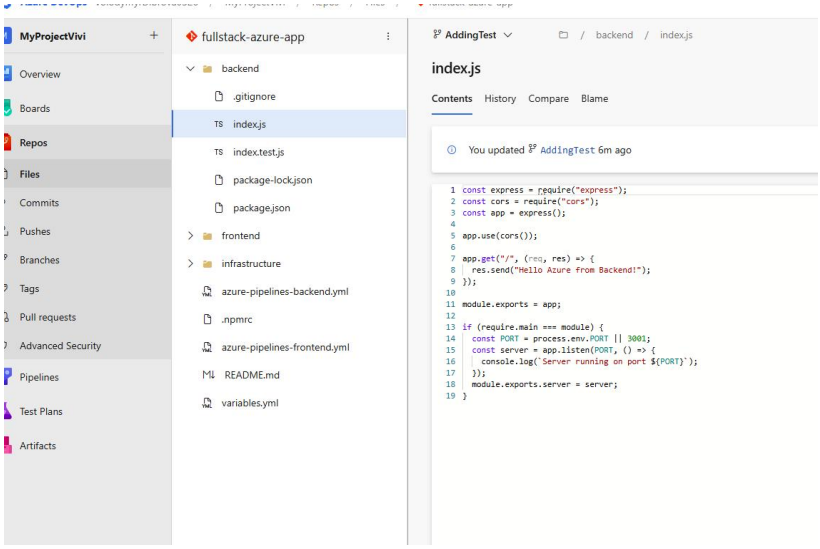

MyProjectVivi
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
Library
Task groups
Deployment groups
Test Plans
Artifacts
Jobs in run #20250222.6
fullstack-azure-app
Jobs
Job 3m 0s
Initialize job <1s
Checkout fullstack-azure-app@m... 2s
Install Node.js <1s
Check if package.json exists bef... <1s
Install dependencies and build fr... 20s
Check build output <1s
Copy .npmrc to frontend <1s
Test package creation 2s
Publish to Azure Artifacts 3s
Install zip utility 2s
Archive frontend build <1s
Install/Update Azure CLI on self-h... 8s
Deploy infrastructure with Bi... 1m 14s
Deploy Frontend to Azure 1m 4s
Post-Job: Checkout fullstack-azu... <1s
Finalize Job <1s
Report build status <1s
Deploy Frontend to Azure
View raw log
Starting: Deploy Frontend to Azure
Task : Azure Web App
Description : Deploy an Azure Web App for Linux or Windows
Version : 1.252.1
Author : Microsoft Corporation
Help : https://aka.ms/azurewebappcroubleshooting
Got service connection details for Azure App Service:'helloazure-frontend'
Updating App Service Application settings. Data: {"WEBSITE_RUN_FROM_PACKAGE":"1"}
Deleting App Service Application settings. Data: ["WEBSITE_RUN_FROM_ZIP"]
Updated App Service Application settings and Kudu Application settings.
Package deployment using ZIP Deploy Initiated.
Successfully deployed web package to App Service.
NOTE: Run From Package makes wwwroot read-only, so you will receive an error when writing files to this directory.
Successfully updated App Service configuration details
App Service Application URL: http://helloazure-frontend.azurewebsites.net
Finishing: Deploy Frontend to Azure

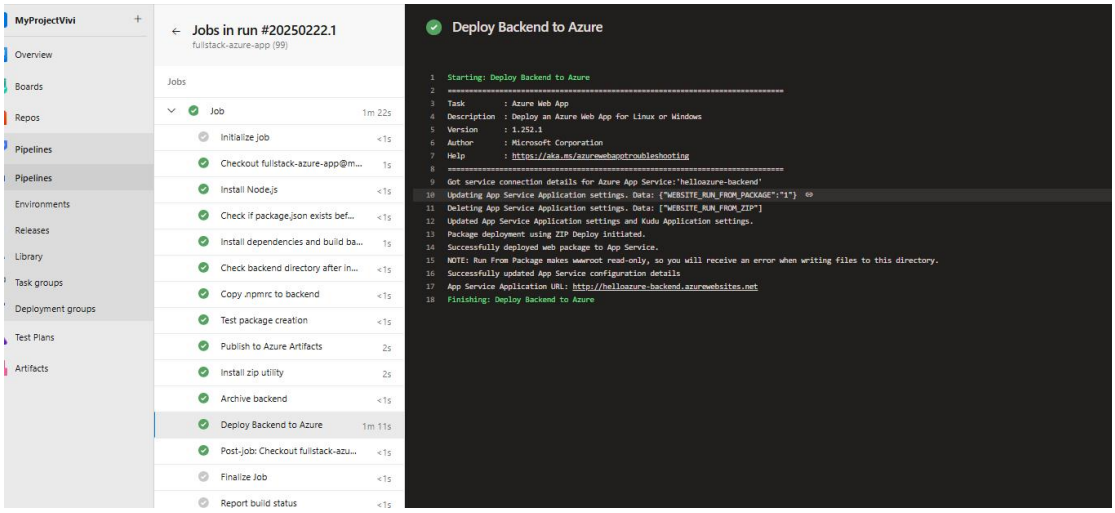

Не защищено
helloazure-frontend.azurewebsites.net
Azure
Hello Azure!

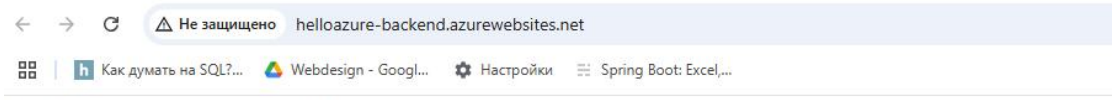

Microsoft Azure
Search resources, services, and docs (G+/)
Home > helloazure-frontend
helloazure-frontend | Deployment Center
Web App
Search
Save
Discard
Browse
Manage publish profile
Sync
Leave Feedback
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Microsoft Defender for Cloud
Events (preview)
Recommended services (preview)
Deployment
Deployment slots
Deployment Center
Settings
Environment variables
Configuration
Authentication
Identity
Backups
Settings
Logs
FTPS credentials
Deploy and build code from your preferred source and build provider. Learn more
Your app is configured with Azure Pipelines as a build provider. Go to the Azure DevOps portal to manage your deployments.
Source
Azure Repos
Disconnect
Azure Repos
Project
MyProjectVivi
Repository
fullstack-azure-app
Branch
refs/heads/main
Build
Build provider
Azure Pipelines

backend

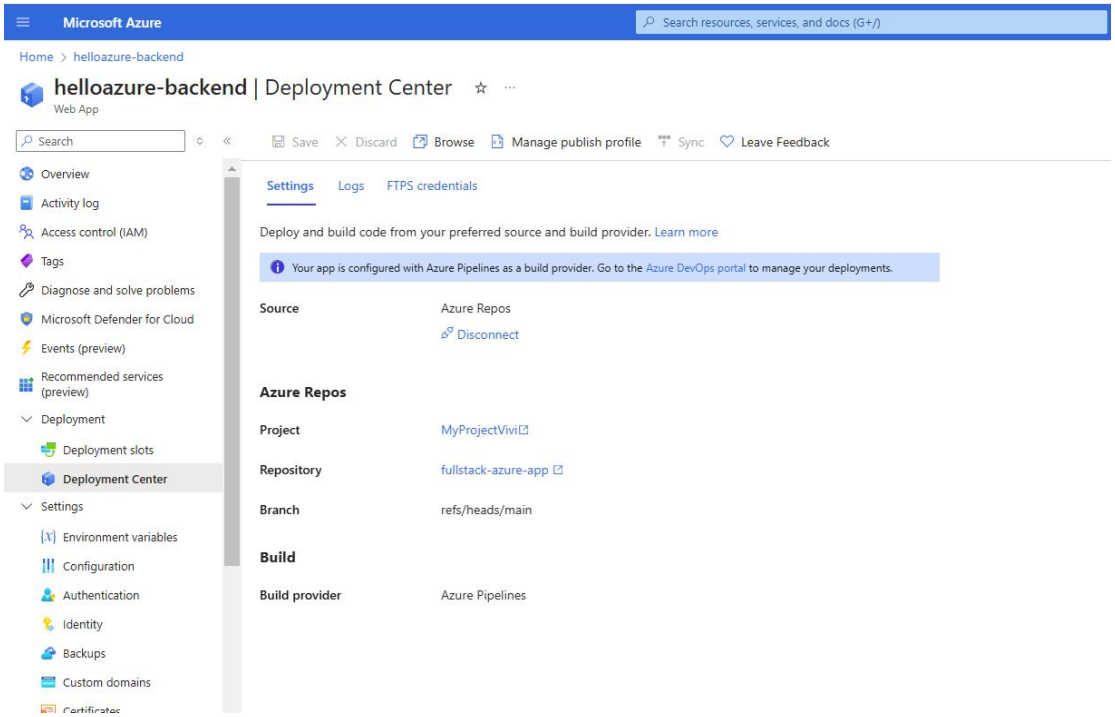

MyProjectVivi
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags
Pull requests
Advanced Security
Pipelines
Test Plans
Artifacts
fullstack-azure-app
backend
frontend
infrastructure
azure-pipelines-backend.yml
.npmrc
azure-pipelines-frontend.yml
README.md
variables.yml
main
azure-pipelines-backend.yml
Contents
History
Compare
Blame


MyProjectVivi
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags
Pull requests
Advanced Security
Pipelines
Test Plans
Artifacts
fullstack-azure-app
backend
.gitignore
index.js
index.test.js
package-lock.json
package.json
frontend
infrastructure
azure-pipelines-backend.yml
.npmrc
azure-pipelines-frontend.yml
README.md
variables.yml
AddingTest
backend
index.js
index.js
Contents
History
Compare
Blame
You updated AddingTest 6m ago
const express = require("express");
const cors = require("cors");
const app = express();
app.use(cors());
app.get("/", (req, res) => {
res.send("Hello Azure from Backend!");
});
module.exports = app;
if (require.main === module) {
const PORT = process.env.PORT || 3001;
const server = app.listen(PORT, () => {
console.log(`Server running on port ${PORT}`);
});
module.exports.server = server;
}



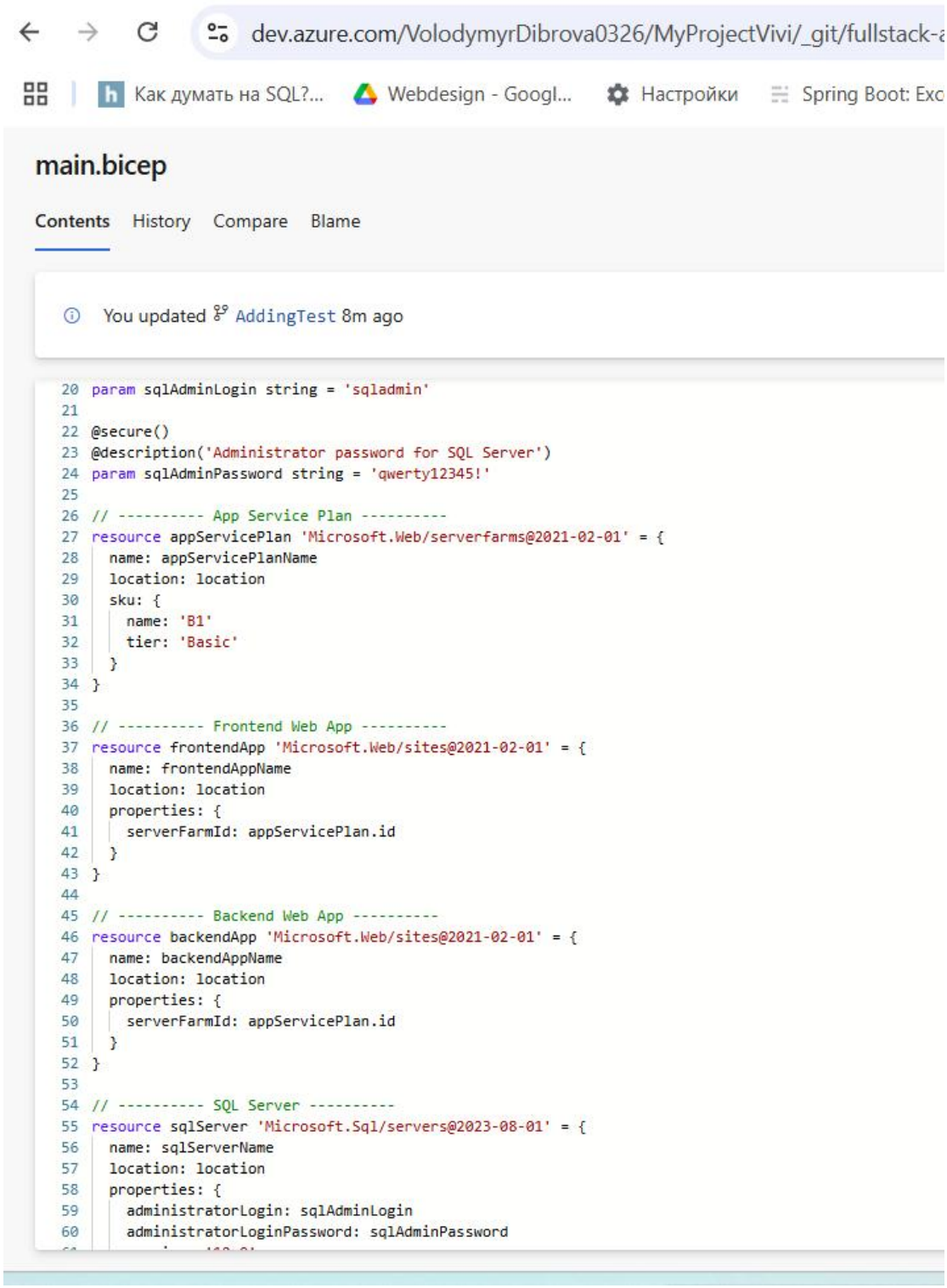


You do not have permission to view this directory or page.



Bicep/ Terraform

bicep


dev.azure.com/VolodymyrDibrova0326/MyProjectVivi/_git/fullstack-a
Как думать на SQL?...
Webdesign - Googl...
Настройки
Spring Boot: Exc
main.bicep
Contents
History
Compare
Blame
You updated AddingTest 8m ago
20 param sqlAdminLogin string = 'sqladmin'
21
22 @secure()
23 @description('Administrator password for SQL Server')
24 param sqlAdminPassword string = 'qwerty12345!'
25
26 // ---------- App Service Plan ----------
27 resource appServicePlan 'Microsoft.Web/serverfarms@2021-02-01' = {
28 name: appServicePlanName
29 location: location
30 sku: {
31 name: 'B1'
32 tier: 'Basic'
33 }
34 }
35
36 // ---------- Frontend Web App ----------
37 resource frontendApp 'Microsoft.Web/sites@2021-02-01' = {
38 name: frontendAppName
39 location: location
40 properties: {
41 serverFarmId: appServicePlan.id
42 }
43 }
44
45 // ---------- Backend Web App ----------
46 resource backendApp 'Microsoft.Web/sites@2021-02-01' = {
47 name: backendAppName
48 location: location
49 properties: {
50 serverFarmId: appServicePlan.id
51 }
52 }
53
54 // ---------- SQL Server ----------
55 resource sqlServer 'Microsoft.Sql/servers@2023-08-01' = {
56 name: sqlServerName
57 location: location
58 properties: {
59 administratorLogin: sqlAdminLogin
60 administratorLoginPassword: sqlAdminPassword

Terraform

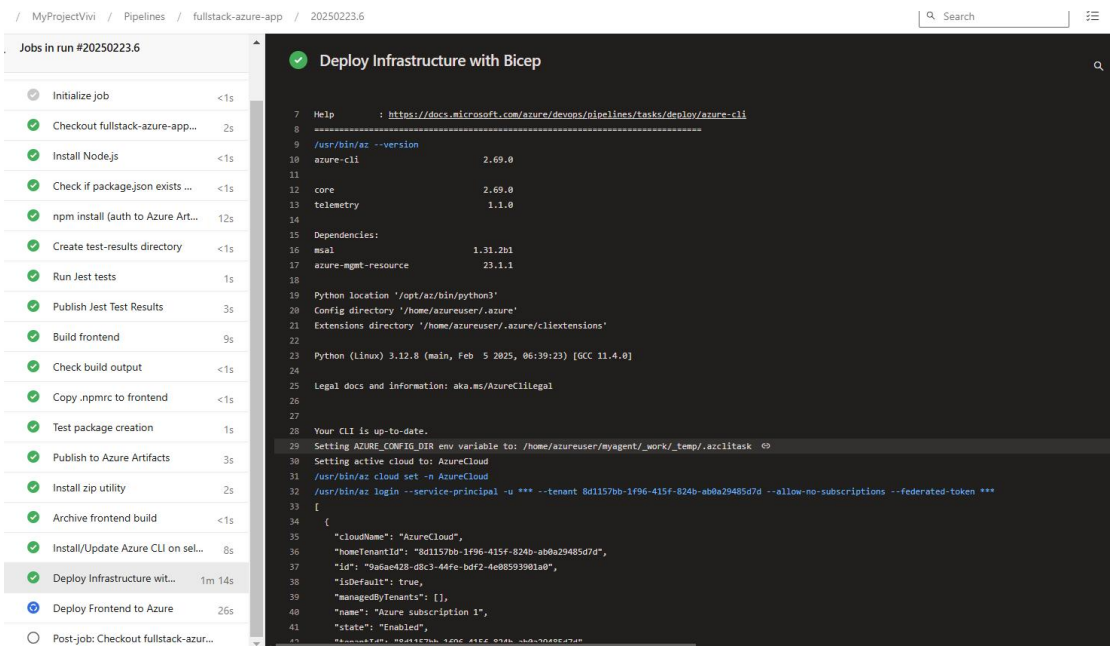
main.tf

Contents History Compare Blame

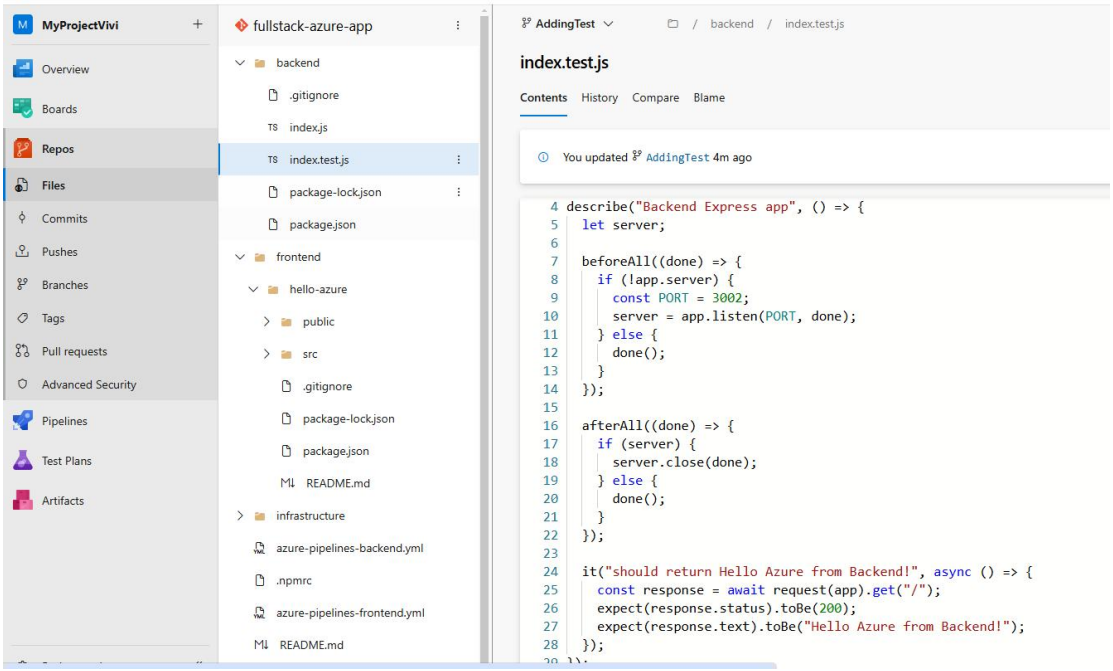
You updated AddingTest 9m ago

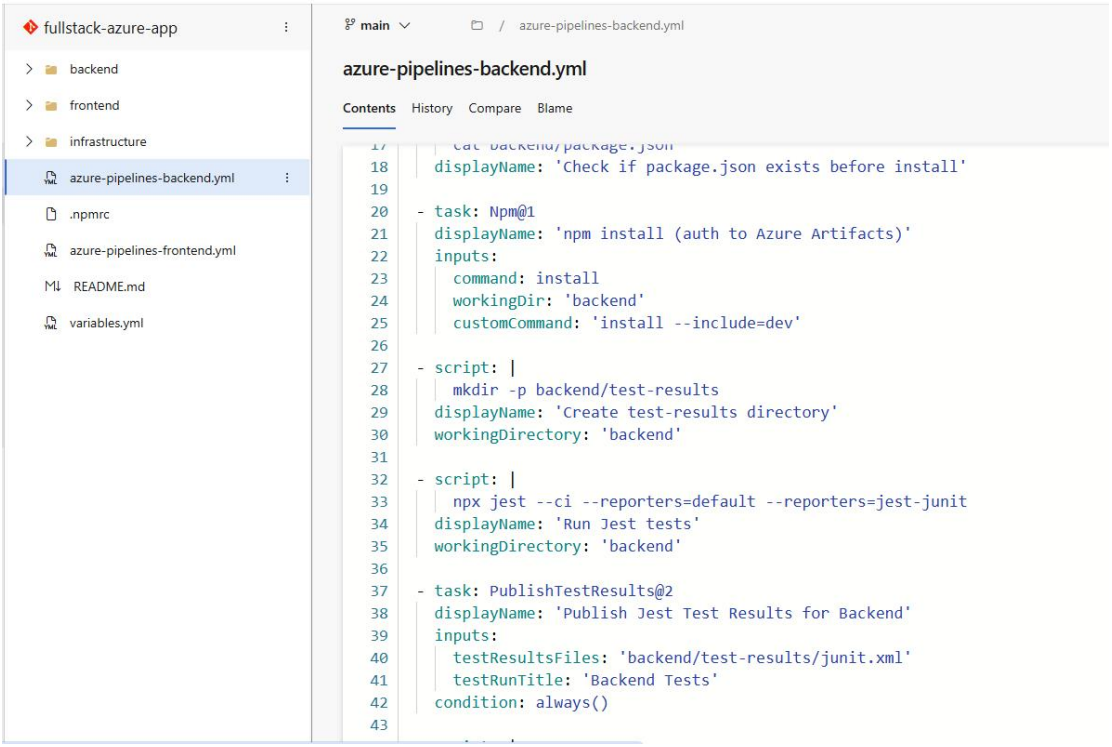
```hcl
resource "azurerm_resource_group" "helloazure" {
  name     = "helloazure-rg"
  location = "West Europe"
}

resource "azurerm_app_service_plan" "plan" {
  name                = "helloazure-plan"
  location            = azurerm_resource_group.helloazure.location
  resource_group_name = azurerm_resource_group.helloazure.name
  kind                = "Linux"
  reserved            = true

  sku {
    tier = "Basic"
    size = "B1"
  }
}

resource "azurerm_app_service" "frontend" {
  name                = "helloazure-frontend"
  location            = azurerm_resource_group.helloazure.location
  resource_group_name = azurerm_resource_group.helloazure.name
  app_service_plan_id = azurerm_app_service_plan.plan.id
}

resource "azurerm_app_service" "backend" {
  name                = "helloazure-backend"
  location            = azurerm_resource_group.helloazure.location
  resource_group_name = azurerm_resource_group.helloazure.name
  app_service_plan_id = azurerm_app_service_plan.plan.id
}

resource "azurerm_sql_server" "sqlserver" {
  name                         = "helloazure-sqlsrv"
  resource_group_name          = azurerm_resource_group.helloazure.name
  location                     = azurerm_resource_group.helloazure.location
  version                      = "12.0"
  administrator_login          = "sqladmin"
  administrator_login_password = "My12345"
  public_network_access        = "Enabled"

```

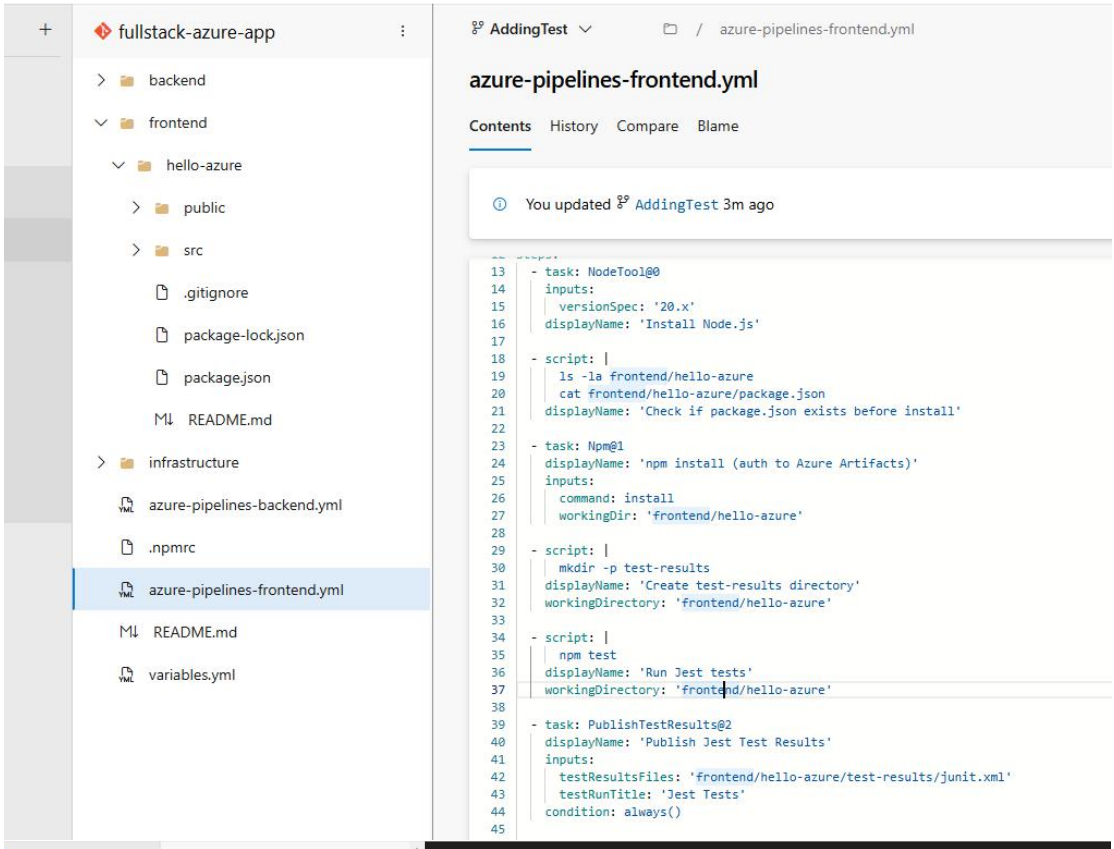
Successfully deployed bicep

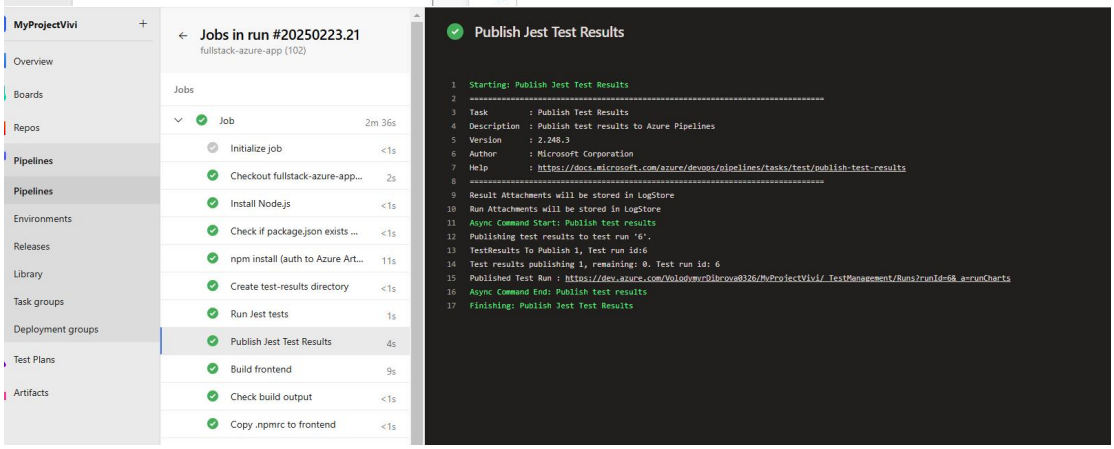


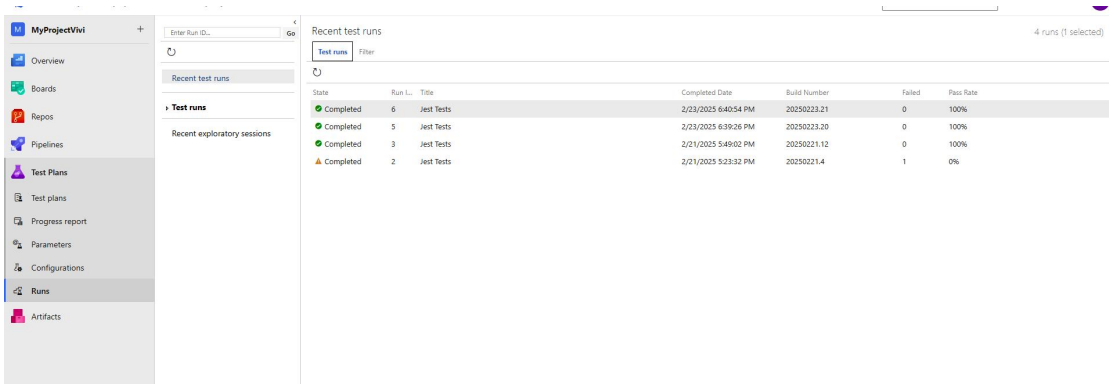
Test:


MyProjectVivi
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags
Pull requests
Advanced Security
Pipelines
Test Plans
Artifacts
fullstack-azure-app
backend
.gitignore
index.js
index.test.js
package-lock.json
package.json
frontend
hello-azure
public
src
.gitignore
package-lock.json
package.json
README.md
infrastructure
azure-pipelines-backend.yml
.npmrc
azure-pipelines-frontend.yml
README.md
AddingTest
backend / index.test.js
index.test.js
Contents History Compare Blame
You updated AddingTest 4m ago
describe("Backend Express app", () => {
let server;
beforeAll((done) => {
if (!app.server) {
const PORT = 3002;
server = app.listen(PORT, done);
} else {
done();
}
});
afterAll((done) => {
if (server) {
server.close(done);
} else {
done();
}
});
it("should return Hello Azure from Backend!", async () => {
const response = await request(app).get("/");
expect(response.status).toBe(200);
expect(response.text).toBe("Hello Azure from Backend!");
});


fullstack-azure-app
backend
frontend
infrastructure
azure-pipelines-backend.yml
.npmrc
azure-pipelines-frontend.yml
README.md
variables.yml
main
azure-pipelines-backend.yml
azure-pipelines-backend.yml
Contents History Compare Blame
displayName: 'Check if package.json exists before install'
- task: Npm@1
displayName: 'npm install (auth to Azure Artifacts)'
inputs:
command: install
workingDir: 'backend'
customCommand: 'install --include=dev'
- script: |
mkdir -p backend/test-results
displayName: 'Create test-results directory'
workingDirectory: 'backend'
- script: |
npx jest --ci --reporters=default --reporters=jest-junit
displayName: 'Run Jest tests'
workingDirectory: 'backend'
- task: PublishTestResults@2
displayName: 'Publish Jest Test Results for Backend'
inputs:
testResultsFiles: 'backend/test-results/junit.xml'
testRunTitle: 'Backend Tests'
condition: always()


fullstack-azure-app
backend
frontend
hello-azure
public
src
.gitignore
package-lock.json
package.json
README.md
infrastructure
azure-pipelines-backend.yml
.npmrc
azure-pipelines-frontend.yml
README.md
variables.yml
AddingTest
azure-pipelines-frontend.yml
azure-pipelines-frontend.yml
Contents
History
Compare
Blame
You updated AddingTest 3m ago
- task: NodeTool@0
inputs:
versionSpec: '20.x'
displayName: 'Install Node.js'
- script: |
ls -la frontend/hello-azure
cat frontend/hello-azure/package.json
displayName: 'Check if package.json exists before install'
- task: Npm@1
displayName: 'npm install (auth to Azure Artifacts)'
inputs:
command: install
workingDir: 'frontend/hello-azure'
- script: |
mkdir -p test-results
displayName: 'Create test-results directory'
workingDirectory: 'frontend/hello-azure'
- script: |
npm test
displayName: 'Run Jest tests'
workingDirectory: 'frontend/hello-azure'
- task: PublishTestResults@2
displayName: 'Publish Jest Test Results'
inputs:
testResultsFiles: 'frontend/hello-azure/test-results/junit.xml'
testRunTitle: 'Jest Tests'
condition: always()


MyProjectVivi
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
Library
Task groups
Deployment groups
Test Plans
Artifacts
Jobs in run #20250223.21
fullstack-azure-app (102)
Jobs
Job 2m 36s
Initialize job <1s
Checkout fullstack-azure-app... 2s
Install Node.js <1s
Check if package.json exists ... <1s
npm install (auth to Azure Art... 11s
Create test-results directory <1s
Run Jest tests 1s
Publish Jest Test Results 4s
Build frontend 9s
Check build output <1s
Copy .npmrc to frontend <1s
Publish Jest Test Results
Starting: Publish Jest Test Results
Task : Publish Test Results
Description : Publish test results to Azure Pipelines
Version : 2.248.3
Author : Microsoft Corporation
Help : https://docs.microsoft.com/azure/devops/pipelines/tasks/test/publish-test-results
Result Attachments will be stored in LogStore
Run Attachments will be stored in LogStore
Async Command Start: Publish test results
Publishing test results to test run '6'.
TestResults To Publish 1, Test run id:6
Test results publishing 1, remaining: 0. Test run id: 6
Published Test Run : https://dev.azure.com/VologdymyrDibrova0326/MyProjectVivi/_TestManagement/Runs?runId=6&_a=runCharts
Async Command End: Publish test results
Finishing: Publish Jest Test Results


MyProjectVivi
Overview
Boards
Repos
Pipelines
Test Plans
Test plans
Progress report
Parameters
Configurations
Runs
Artifacts
Recent test runs
Test runs
Recent exploratory sessions
Recent test runs
4 runs (1 selected)
State Run I... Title Completed Date Build Number Failed Pass Rate
Completed 6 Jest Tests 2/23/2025 6:40:54 PM 20250223.21 0 100%
Completed 5 Jest Tests 2/23/2025 6:39:26 PM 20250223.20 0 100%
Completed 3 Jest Tests 2/21/2025 5:49:02 PM 20250221.12 0 100%
Completed 2 Jest Tests 2/21/2025 5:23:32 PM 20250221.4 1 0%

Monitoring without simulation